

CHRISTOFER SCHWARTZ



**FERRAMENTA DE PREVISÃO DE DESEMPENHO DO  
SISTEMA ISDB-T DE TV DIGITAL**

Dissertação apresentada como requisito parcial  
à obtenção do grau de Mestre. Programa de  
Pós-Graduação em Engenharia Elétrica, Setor  
de Tecnologia, Universidade Federal do Paraná.

Orientador: Prof. Dr. Evelio Martín García  
Fernández

CURITIBA

2011

CHRISTOFER SCHWARTZ

**FERRAMENTA DE PREVISÃO DE DESEMPENHO DO  
SISTEMA ISDB-T DE TV DIGITAL**

Dissertação apresentada como requisito parcial  
à obtenção do grau de Mestre. Programa de  
Pós-Graduação em Engenharia Elétrica, Setor  
de Tecnologia, Universidade Federal do Paraná.

Orientador: Prof. Dr. Evelio Martín García  
Fernández

CURITIBA

2011

Schwartz, Christofer

Ferramenta de previsão de desempenho do sistema ISDB-T de TV digital / Christofer Schwartz. – Curitiba, 2011.  
118 f.: il., tabs.

Orientador: Prof. Dr. Evelio Martín García Fernández

Dissertação (Mestrado) – Dissertação apresentada como requisito Parcial à obtenção do grau de mestre. Programa de Pós-Graduação em Engenharia Elétrica, Setor de Tecnologia, Universidade Federal do Paraná.

Inclui Bibliografia.

1. Transmissão digital. 2. Comunicações digitais. I. Título. II. Fernández, Evelio Martín García. III. Universidade Federal do Paraná.

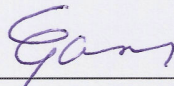
CDD 621.38275

CHRISTOFER SCHWARTZ

**FERRAMENTA DE PREVISÃO DE DESEMPENHO DO  
SISTEMA ISDB-T DE TV DIGITAL**

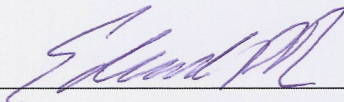
Dissertação aprovada como requisito parcial à obtenção do grau de Mestre no Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Paraná, pela comissão formada pelos professores:

**Banca Examinadora**



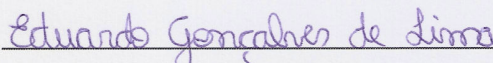
---

Prof. Dr. Everaldo Martín García Fernández  
Universidade Federal do Paraná  
Orientador



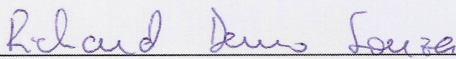
---

Prof. Dr. Eduardo Parente Ribeiro  
Universidade Federal do Paraná  
Convidado



---

Prof. Ph.D. Eduardo Gonçalves de Lima  
Universidade Federal do Paraná  
Convidado



---

Prof. Dr. Richard Demo Souza  
Universidade Tecnológica Federal do Paraná  
Convidado

Curitiba, 25 de Fevereiro de 2011

CHRISTOFER SCHWARTZ

**FERRAMENTA DE PREVISÃO DE DESEMPENHO DO  
SISTEMA ISDB-T DE TV DIGITAL**

Dissertação aprovada como requisito parcial à obtenção do grau de Mestre no Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Paraná, pela comissão formada pelos professores:

**Banca Examinadora**

---

Prof. Dr. Evelio Martín García Fernández  
Universidade Federal do Paraná  
Orientador

---

Prof. Dr. Eduardo Parente Ribeiro  
Universidade Federal do Paraná  
Convidado

---

Prof. Ph.D. Eduardo Gonçalves de Lima  
Universidade Federal do Paraná  
Convidado

---

Prof. Dr. Richard Demo Souza  
Universidade Tecnológica Federal do Paraná  
Convidado

Curitiba, 25 de Fevereiro de 2011

## *Agradecimentos*

*A Deus, acima de tudo.*

*Ao meu orientador, Prof. Dr. Evelio Martín García Fernández pelas suas importantes contribuições, dedicação, paciência e ajuda.*

*A meus pais, Carlos Alfredo Schwartz e Carmen Aparecida Bet Schwartz, bem como a meu irmão Stephan Schwartz, por suas lições de vida, amor, suporte, trabalho e dedicação.*

*A todos os meus amigos e colegas da UFPR, e em especial ao Rudolfo Runcos, Carlos Ignácio de Mattos, Rodrigo Seger, Roberto Krauss e Camilo Coelho, pelo apoio e suporte pessoal.*

*Ao corpo docente do Departamento de Engenharia Elétrica da UFPR, pelo aprendizado.*

*À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes).*

*Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ).*

---

## CONTEÚDO

---

<b>LISTA DE FIGURAS</b>	<b>x</b>
<b>LISTA DE TABELAS</b>	<b>xi</b>
<b>RESUMO</b>	<b>xix</b>
<b>ABSTRACT</b>	<b>xx</b>
<b>1 INTRODUÇÃO</b>	<b>1</b>
1.1 Motivação e objetivo . . . . .	4
1.2 Trabalhos relacionados . . . . .	6
1.3 Estrutura da dissertação . . . . .	8
<b>2 REVISÃO DE CONCEITOS</b>	<b>10</b>
2.1 Introdução . . . . .	10
2.2 Sistemas multiportadoras . . . . .	11

2.3	O sistema OFDM . . . . .	14
2.3.1	Discretização . . . . .	16
2.3.2	Intervalo de guarda . . . . .	17
2.3.3	Sincronismo . . . . .	19
2.4	O canal de transmissão . . . . .	20
2.4.1	Modelagem . . . . .	24
2.5	Considerações finais . . . . .	25
<b>3</b>	<b>O SISTEMA ISDB-T</b>	<b>26</b>
3.1	Introdução . . . . .	26
3.2	Remultiplexador . . . . .	28
3.3	Codificação externa . . . . .	29
3.4	Dispersor de energia . . . . .	31
3.5	Ajuste de atrasos . . . . .	32
3.6	Codificador interno . . . . .	32
3.7	Mapeadores . . . . .	35
3.7.1	Deteção coerente e não-coerente . . . . .	36
3.8	Estratégias de decisão dos demoduladores . . . . .	38
3.9	O quadro OFDM . . . . .	40
3.9.1	Modos de operação . . . . .	40
3.9.2	Estrutura do quadro OFDM . . . . .	40
3.9.3	Divisão hierárquica . . . . .	41
3.9.4	Sinais pilotos . . . . .	42



3.10	Entrelaçadores . . . . .	45
3.10.1	Bloco de entrelaçamento de byte . . . . .	46
3.10.2	Bloco de entrelaçamento de bit . . . . .	47
3.10.3	Bloco de entrelaçamento no tempo . . . . .	48
3.10.4	Bloco de entrelaçamento na frequência . . . . .	49
3.11	Considerações finais . . . . .	50
<b>4</b>	<b>METODOLOGIA DO SIMULADOR</b>	<b>51</b>
4.1	Introdução . . . . .	51
4.2	Considerações feitas a respeito do ISDB-T . . . . .	53
4.3	Fontes de dados para transmissão . . . . .	54
4.3.1	Dados gerados pelo simulador . . . . .	55
4.3.2	Dados de arquivos externos . . . . .	57
4.4	Estrutura do simulador . . . . .	58
4.5	Modelos de canais utilizados . . . . .	72
4.6	Estimação de canal . . . . .	74
4.7	Considerações finais . . . . .	76
<b>5</b>	<b>RESULTADOS</b>	<b>77</b>
5.1	Introdução . . . . .	77
5.2	Validação . . . . .	77
5.3	Bloco de estimação de canal . . . . .	80
5.4	Curvas de desempenho . . . . .	86
5.5	Resultados complementares . . . . .	90

5.5.1	Figura da distribuição de segmentos e intervalo de guarda . . . . .	91
5.5.2	Figura dos TSPs remultiplexados . . . . .	91
5.5.3	Imagens transmitidas . . . . .	92
5.5.4	Vídeos das constelações . . . . .	93
5.6	Considerações finais . . . . .	94
<b>6</b>	<b>O SOFTWARE CRIADO</b>	<b>95</b>
6.1	Introdução . . . . .	95
6.2	Estrutura geral . . . . .	96
6.3	O ISDB-T Analyser . . . . .	98
6.4	Vantagens e Desvantagens . . . . .	107
6.5	Considerações finais . . . . .	109
<b>7</b>	<b>CONCLUSÕES</b>	<b>111</b>
7.1	Sugestões para trabalhos futuros . . . . .	113
	<b>BIBLIOGRAFIA</b>	<b>118</b>

---

## Lista de Figuras

---

1.1	Distribuição dos padrões de TV Digital no mundo. . . . .	2
2.1	Exemplo de um sistema de comunicação digital. . . . .	11
2.2	Espectro de frequência de um sinal modulado por portadora única. . . . .	12
2.3	Espectro de frequência de um sinal multiportadoras. . . . .	12
2.4	Espectro OFDM para 16 subportadoras. . . . .	15
2.5	Modulador OFDM simples. . . . .	16
2.6	Transmissor OFDM com IFFT/IDFT. . . . .	17
2.7	Ilustração da necessidade de inserção de prefixo cíclico. . . . .	18
2.8	Ilustração da inserção do intervalo de guarda ao símbolo OFDM. . . . .	19
2.9	Variações temporais de propagação por multipercurso. . . . .	21
2.10	Diagrama de blocos de configuração de ruídos do canal. . . . .	24
3.1	Visão macro do sistema de transmissão ISDB-T. . . . .	27

3.2	Diagrama de blocos do sistema de transmissão do ISDB-T <sub>b</sub> . . . . .	28
3.3	Estrutura de um TSP do sistema ISDB-T. . . . .	29
3.4	Diagrama de bloco das classes dos códigos corretores de erros. . . . .	30
3.5	Estrutura de dispersão de energia. . . . .	31
3.6	Exemplo de um codificador convolucional de $k = 1, n = 1$ e $m = 2$ . . . . .	33
3.7	Estrutura de codificação e punctionamento do código convolucional utilizado no sistema ISDB-T. . . . .	34
3.8	Comparativo das constelações utilizadas pelo sistema ISDB-T. . . . .	36
3.9	Desempenho das modulações do sistema ISDB-T em canais AWGN. . . . .	37
3.10	Estrutura de um quadro OFDM. . . . .	41
3.11	Ilustração do espectro de frequências dos exemplos de distribuição de seg- mentos às camadas hierárquicas. (a) Configuração 1, (b) Configuração 2 e (c) Configuração 3. . . . .	42
3.12	Circuito de geração de dados das SPs. . . . .	43
3.13	Exemplo de distribuição das portadoras de controle em um segmento OFDM modo 1. . . . .	44
3.14	Diagrama ilustrativo de um entrelaçador de bloco. . . . .	45
3.15	Ilustração do entrelaçador convolucional de $T_{ent} = 3$ e $\Delta = 1$ . . . . .	46
3.16	Estruturas de entrelaçamento de bytes do sistema ISDB-T. . . . .	47
3.17	Estruturas de entrelaçamento de bits do sistema ISDB-T. (a) Modulação DQPSK, (b) Modulação QPSK, (c) Modulação 16-QAM e (d) Modulação 64-QAM. . . . .	48
3.18	Localização das subportadoras de dados nos símbolos OFDM após o en- treçamento no tempo para $I = 8$ e $Modo = 1$ . . . . .	49

4.1	Blocos do transmissor implementados em Matlab. . . . .	55
4.2	Blocos do receptor implementados em Matlab. . . . .	56
4.3	Dados importados de arquivos externos. . . . .	57
4.4	Estrutura de dados de um arquivo de texto externo. . . . .	58
4.5	Fluxograma do simulador (parte 1). . . . .	59
4.6	Estrutura de um segmento OFDM em modo 1, com modulação diferencial (60 símbolos OFDM e 96 subportadoras). . . . .	61
4.7	Estrutura de um segmento OFDM em modo 1, com modulação síncrona (60 símbolos OFDM e 96 subportadoras). . . . .	61
4.8	Estrutura do quadro OFDM em modo 1, configuração 3 e segmento central com modulação diferencial. . . . .	62
4.9	Fluxograma do simulador (parte 2). . . . .	64
4.10	Exemplo de relação entre $E_b/N_0$ para as três camadas com diferentes parâmetros. . . . .	67
4.11	Fluxograma do simulador (parte 3). . . . .	69
4.12	Informações de nível de ruído para cada camada hierárquica. . . . .	71
4.13	Representação do código de simulação. . . . .	72
4.14	Quantidade de portadoras SPs utilizadas para estimação de canal. (a) Estimações em uma dimensão e (b) Estimações em duas dimensões. . . . .	74
5.1	Curvas de desempenho para as diferentes modulações com CC de $taxa =$ $3/4$ , e CC de $taxa = 3/4$ concatenado com RS(204,188), para canais AWGN. (a) Curvas para comparação e (b) Curvas obtidas com o simu- lador criado. . . . .	78

5.2	Curva de desempenho para as diferentes modulações com CC de $\text{taxa} = 3/4$ , e CC de $\text{taxa} = 3/4$ concatenado com RS(204,188), para canais AWGN.	79
5.3	Curva de desempenho para modulação 64-QAM com CC de $\text{taxa} = 3/4$ concatenado com RS(204,188), para canal Brasil A. (a) Curva para comparação e (b) Curva obtida com o simulador criado. . . . .	80
5.4	Constelação 16-QAM recebida de 1 símbolo OFDM sem estimação, corrompida por um canal de desvanecimento Rayleigh (Brasil B). . . . .	81
5.5	Resultado do estimador de canal 1D <i>Cubic-Spline</i> para 1 símbolo OFDM em modo 1 (Canal Brasil B). . . . .	82
5.6	Resultado do estimador de canal 1D <i>Cubic-Spline</i> para as 50 primeiras subportadoras de 1 símbolo OFDM em modo 1 (Canal Brasil B). . . . .	82
5.7	Constelação 16-QAM das subportadoras de 1 símbolo OFDM com estimação 1D <i>Cubic-Spline</i> corrompida por um canal de desvanecimento Rayleigh (Brasil B). . . . .	83
5.8	Resultado do estimador de canal 2D <i>Cubic-Spline</i> para 1 símbolo OFDM em modo 1 (Canal Brasil B). . . . .	84
5.9	Resultado do estimador de canal 2D <i>Cubic-Spline</i> para as 50 primeiras subportadoras de 1 símbolo OFDM em modo 1 (Canal Brasil B). . . . .	84
5.10	Constelação 16-QAM das subportadoras de 1 símbolo OFDM com estimação 2D <i>Cubic-Spline</i> corrompida por um canal de desvanecimento Rayleigh (Brasil B). . . . .	85
5.11	Constelação 16-QAM das subportadoras de 1 símbolo OFDM com estimação 2D <i>Cubic-Spline</i> corrompida por um canal de desvanecimento Rayleigh (Brasil B) + 12dB de ruído AWGN. . . . .	85
5.12	Queda de desempenho nas constelações decorrentes de erros de estimação em canais com desvanecimento + ruído AWGN. . . . .	86

5.13	Exemplo de desempenho de uma transmissão em HDTV. . . . .	88
5.14	Exemplo de desempenho de uma transmissão em EDTV. . . . .	89
5.15	Exemplo de desempenho de uma transmissão em SDTV. . . . .	89
5.16	Ilustração da distribuição dos segmentos OFDM nas camadas hierárquicas para a configuração 2 e inserção do intervalo de guarda. . . . .	91
5.17	TS remultiplexado com os dados das imagens TIFF para a configuração 3.	92
5.18	Exemplo de imagem recebida em 3 quadros OFDM, configuração 1, modo 3, modulação QPSK, intervalo de guarda de 1/8 e código convolucional puncionado para 3/4. . . . .	93
6.1	Fluxograma simplificado do software de visualização de resultados. . . . .	97
6.2	Janela de <i>splash</i> do software de visualização de resultados. . . . .	98
6.3	Janela de importação de simulações do software de visualização de resultados.	99
6.4	Janela principal do software de visualização de resultados. . . . .	100
6.5	Janela de principal após seleção dos parâmetros e conclusão da busca. . . .	101
6.6	Janela de visualização de imagens recebidas do software de visualização de resultados. . . . .	102
6.7	Janela de geração do vídeo das constelações recebidas do software de visu- alização de resultados. . . . .	103
6.8	Janela de visualização da distribuição de segmentos do quadro OFDM e adição do intervalo de guarda do software visualização de resultados. . . .	104
6.9	Janela de visualização da estrutura do TS remultiplexado do software de visualização de resultados. . . . .	105
6.10	Consulta ao botão de “ajuda rápida” da janela de descrição de portadoras.	106

6.11 Janela principal do editor de informações de banco de dados do software de visualização de resultados. . . . .	107
--	-----



---

## Lista de Tabelas

---

1.1	Parâmetros de transmissão do sistema ISDB-T . . . . .	4
3.1	Exemplos de distribuição de segmento entre as camadas hierárquicas. . . .	41
3.2	Quantidade de portadoras de controle para um símbolo OFDM. . . . .	44
4.1	Especificação de atraso e atenuação dos modelos de canais Brasil. . . . .	73
5.1	Áreas de cobertura para o sistema ISDB-T. . . . .	90

---

## Lista de Siglas e Abreviaturas

---

ISDB-T	<i>Integrated Services Digital Broadcasting Terrestrial</i>
ATSC	<i>Advanced Television System Committee</i>
DVB	<i>Digital Video Broadcasting</i>
ISDB	<i>Integrated Service Digital Broadcasting</i>
DMB	<i>Digital Multimedia Broadcast</i>
8-VSB	<i>Eight-Vestigial Side Band</i>
DVB-T	<i>Digital Video Broadcasting Terrestrial</i>
SFN	<i>Single Frequency Network</i>
ARIB	<i>Association of Radio Industries and Businesses</i>
BST-OFDM	<i>Band Segmented Transmission Orthogonal Frequency Division Multiplexing</i>
OFDM	<i>Orthogonal Frequency Division Multiplexing</i>

DMB-T/H	<i>Digital Multimedia Broadcast - Terrestrial/Handheld</i>
CPqD	Centro de Pesquisa e Desenvolvimento em Telecomunicações
INATEL	Instituto Nacional de Telecomunicações
LDPC	<i>Low Density Parity Check Code</i>
MIMO	<i>Multiple-Input, Multiple-Output</i>
MI-SBTVD	Modulação Inovadora para o Sistema Brasileiro de TV Digital
H.264/AVC	H.264/ <i>Advanced Video Coding</i>
SDTV	<i>Standard Definition Television</i>
EDTV	<i>Enhanced Definition Television</i>
HDTV	<i>High-definition Television</i>
LMS	<i>Least Mean Square</i>
TS	<i>Transport Stream</i>
ISI	<i>Inter-Symbol Interference</i>
FDM	<i>Frequency Division Multiplexing</i>
IDFT	<i>Inverse Discrete Fourier Transform</i>
IFFT	<i>Inverse Fast Fourier Transform</i>
AWGN	<i>Additive White Gaussian Noise</i>
UHF	<i>Ultra High Frequency</i>
VHF	<i>Very High Frequency</i>
QAM	<i>Quadrature Amplitude Modulation</i>

DQPSK	<i>Differential Quadrature Phase Shift Keying</i>
LLR	<i>Log-Likelihood Ratio</i>
SP	<i>Scattered Pilot</i>
CP	<i>Continual Pilot</i>
TMCC	<i>Transmission and Multiplexing Configuration Control</i>
AC	<i>Auxiliary Channel</i>
FIFO	<i>First In First Out</i>
AVI	<i>Audio Video Interleave</i>
DB	<i>Data Base</i>
RAM	<i>Random Access Memory</i>
ITU-R	<i>International Telecommunication Union - Radiocommunication Sector</i>

---

## Lista de Símbolos

---

$N_c$	Quantidade de subportadoras
$B_w$	Largura de banda do canal [Hz]
$R$	Taxa de transmissão desejada [bits/segundo]
$fd_{max}$	Espalhamento Doppler do canal
$T_s$	Duração do símbolo OFDM [s]
$T_d$	Duração de um símbolo da fonte [s]
$F_s$	Espaçamento de frequência das subportadoras ortogonais do sistema OFDM.
$f_n$	Frequência das subportadoras ortogonais do sistema OFDM.
$W_n$	Frequência angular [rad]
$x_v$	Sinal de um símbolo OFDM discretizado.
$R_n$	Saída do demodulador OFDM.
$y_n$	Sinal de entrada, amostrado a uma taxa de $1/T_s$ , do demodulador OFDM.

$\tau_{max}$	Duração da resposta ao impulso do canal
$T_g$	Duração do intervalo de guarda [s].
$T'_s$	Duração do símbolo OFDM após a adição do prefixo cíclico.
$T_c$	Tempo de coerência
$T_{\Delta path}$	Variação do tempo de propagação do sinal com multipercursos
$T_s$	Tempo de transmissão de um símbolo.
$B_w$	Largura de banda.
$h(t)$	Resposta ao impulso do canal
$A_i$	Atenuação do sinal no $i$ -ésimo multipercurso.
$\tau_i$	Atraso do sinal no $i$ -ésimo percurso.
$N_{path}$	Quantidade de caminhos com multipercursos.
$k$	Comprimento da mensagem de entrada codificador (código de blocos).
$n$	Comprimento da palavra código (código de blocos).
$t_{cb}$	Capacidade de correção de erros do código corretor de erros.
$g(x)$	Polinômio gerador do código RS.
$k_{cc}$	Número de entradas do codificador convolucional.
$n_{cc}$	Quantidade de saídas do codificador convolucional.
$m_{cc}$	Quantidade de registros de deslocamento do codificador convolucional.
$R_{cc}$	Taxa do código convolucional.
$k_{bits}$	Quantidade de bits por símbolo de uma constelação.

$M$	Quantidade de símbolos distintos de uma constelação qualquer
$b$	Bit qualquer.
$LLR(b)$	LLR exato de um bit qualquer.
$LLR_{ap}(b)$	LLR aproximado de um bit qualquer.
$S_s$	Número de segmentos OFDM com modulação síncrona.
$S_d$	Número de segmentos OFDM com modulação diferencial.
$S_t$	Número total de segmentos OFDM.
$T_{ent}$	Período do entrelaçador
$I$	Profundidade de entrelaçamento.
$i$	Entrada do entrelaçador.
$\pi(i)$	Saída do entrelaçador.
$C$	Potência média das subportadoras do símbolo OFDM.
$T_{ss}$	Tempo de amostragem na saída da IFFT (após adição do prefixo cíclico).
$FS$	Frequência de amostragem da IFFT/FFT.
$e$	Quantidade de símbolos corrompidos de uma palavra código.
$Y(n_s, n_p)$	O sinal da subportadora transmitida (antes da IFFT)
$X(n_s, n_p)$	O sinal da subportadora recebida (após a FFT)
$H(n_s, n_p)$	É a resposta do canal para uma única subportadora do símbolo OFDM.
$N(n_s, n_p)$	É o ruído AWGN adicionado à uma subportadora
$n_s$	Índice do símbolo OFDM no quadro.

$n_p$  Índice da subportadora (no sentido da frequência) no símbolo OFDM.

$\hat{H}(n_s, n_p)$  Resposta estimada do canal para uma única subportadora do símbolo OFDM.



---

## Resumo

---

Conhecida a importância do sistema televisivo brasileiro, surge a necessidade de realizar transmissões digitais de programas de televisão em território brasileiro. Junto com essa necessidade, vem a obrigação de operar corretamente os transmissores, visando obter o desempenho ótimo do sistema adotado pelo Brasil, o ISDB-T (*Integrated Services Digital Broadcasting Terrestrial*). Tal sistema tem como característica possuir grande flexibilidade para realizar transmissões em diversos tipos de terrenos, tendo vários parâmetros que podem ser ajustados pelo operador.

Esse trabalho tem como objetivo documentar a criação de uma ferramenta de previsão de desempenho para o sistema ISDB-T. Busca-se fazer uma avaliação dos impactos causados pelas diferentes combinações de parâmetros que o sistema pode assumir através de um software. Desta forma, pode-se obter uma melhoria na qualidade da transmissão. Diferentes formas de visualização de resultados são abordadas, como por exemplo, a visualização de curvas de desempenho, visualização de imagens e vídeos transmitidos e a geração de vídeos das constelações recebidas. As várias etapas de tratamento da informação são implementadas em Matlab e documentadas neste trabalho.

**Palavras chave:** ISDB-T, TV Digital, Parâmetros, Desempenho

---

## Abstract

---

Given the importance of brazilian television system, it is necessary to implement digital broadcasting in brazilian territory. In order to achieve optimal performance on the adopted brazilian standard, the ISDB-T, correct transmitters operation is a must. This standard has many configurable parameters, granting it a great flexibility to adapt its transmission to different terrains.

The objective of this work is to document the creation of a performance estimation tool for the ISDB-T standard. This tool evaluates the influence of different ISDB-T parameter configurations through software simulation. This makes easier for an operator to find optimal transmission parameters. Results from simulations are shown in different ways, such as performance curves, transmitted image and video visualization, and received constellation animations. All information treatment steps are implemented in Matlab and described in this work. Also, a database containing results from several simulations using different parameter sets is created. Another software developed in this work uses this database to show the impacts in transmission due to ISDB-T parameter variations.

**Key words:** ISDB-T, Digital TV, Parameters, Performance

# CAPÍTULO 1

---

## Introdução

---

Avanços da eletrônica digital tornaram possível a implementação de sistemas de televisão digital, os quais permitem melhora na qualidade do sinal e adição de novos recursos em relação com o sistema analógico. Entre esses avanços está o desenvolvimento de microprocessadores dedicados que trabalham em frequências da ordem de gigahertz.

Segundo [1], em abril de 2007, cerca de 97,1% das casas brasileiras possuíam algum tipo de televisor, levando este o título de aparelho elétrico mais comumente encontrado nos lares brasileiros. Ele ficou a frente até mesmo da geladeira, que conquistou o segundo lugar com 96% de popularidade. Esses dados expõem a importância da televisão na vida dos brasileiros, justificando a busca por melhorias no sistema televisivo do país.

Na atualidade existem quatro padrões de TV Digital usados no mundo, são eles:

- ATSC (*Advanced Television System Committee*);
- DVB (*Digital Video Broadcasting*);
- ISDB (*Integrated Service Digital Broadcasting*);

- DMB (*Digital Multimedia Broadcast*).

A Figura 1.1 ilustra a distribuição mundial dos padrões listados, em setembro de 2010 [2].

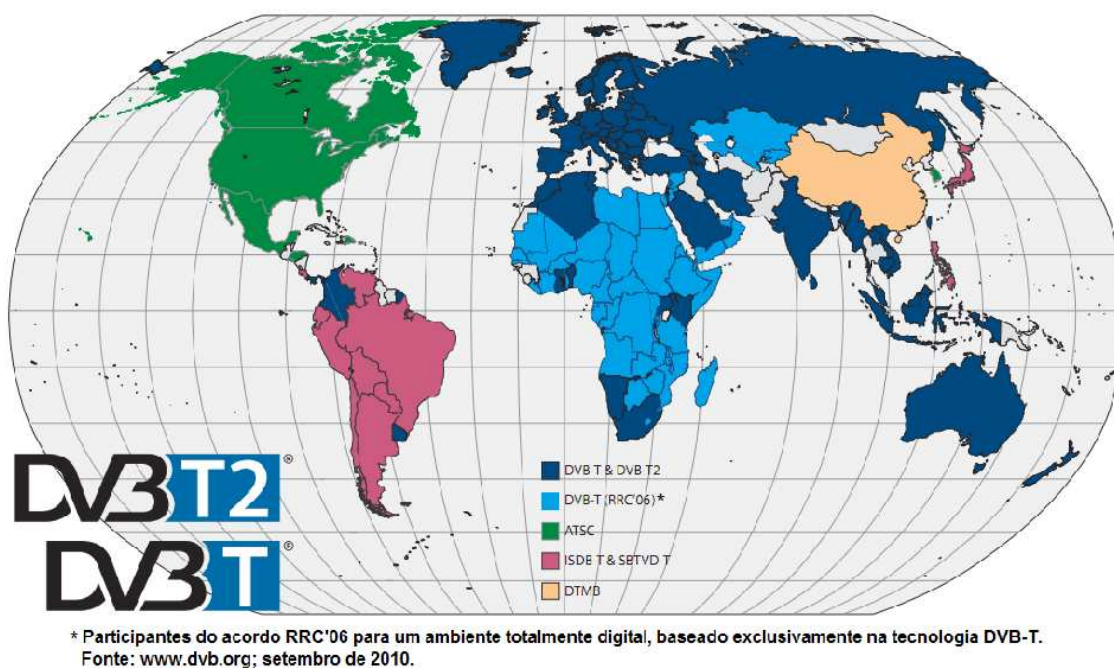


Figura 1.1: Distribuição dos padrões de TV Digital no mundo.

O sistema norte americano ATSC, primeiro padrão de TV Digital desenvolvido, teve como principal objetivo a melhoria na qualidade de som e imagem, não se preocupando com a recepção móvel. Esse padrão utiliza a técnica de modulação vestigial de oito níveis, 8-VSB (*Eight-Vestigial Side Band*) [3].

O sistema DVB-T (*Digital Video Broadcasting Terrestrial*), iniciado em 1993 na Europa, teve atenção voltada para robustez frente a interferências do canal de transmissão. Assim, o comitê criado para a concepção do padrão optou pela técnica COFDM. Outros recursos foram previstos, como a utilização de divisão hierárquica de transmissão e a transmissão através de SFN (*Single Frequency Network*) [2].

O padrão japonês (ISDB-T), desenvolvido pela ARIB (*Association of Radio Industries and Businesses*) em 1998, teve de se preocupar com outro aspecto de transmissão: recepção móvel. Assim, foi utilizado o sistema de modulação BST-OFDM (*Band Seg-*

*mented Transmission Orthogonal Frequency Division Multiplexing*). Nesta variação de configuração OFDM (*Orthogonal Frequency Division Multiplexing*), o espectro é dividido em 13 segmentos que por sua vez são combinados em três camadas hierárquicas, visando recepção parcial do sinal por meio de receptores portáteis. Além da segmentação da banda, o sistema ISDB-T também conta com o entrelaçamento temporal de subportadoras, que visa minimizar as interferências causadas pela recepção móvel e multi-percurso [4].

Segundo [5], em Setembro de 2006, o sistema DMB-T/H (*Digital Multimedia Broadcast - Terrestrial/Handheld*) foi criado devido ao governo chinês ser contra o pagamento de royalties aos países desenvolvedores dos padrões já existentes. Tal padrão é fruto da junção de duas tecnologias desenvolvidas nas universidades de Tsinghua e Shanghai Jiaotong. A universidade de Tsinghua desenvolveu um sistema que utiliza modulação OFDM, similar aos padrões DVB-T e ISDB-T, enquanto o sistema desenvolvido em Jiaotong é um sistema de portadora única, baseado no padrão norte americano [2].

No Brasil foi iniciado um projeto sob a supervisão do CPqD (Centro de Pesquisa e Desenvolvimento em Telecomunicações), tendo o INATEL (Instituto Nacional de Telecomunicações) como proponente e como parceiros a Universidade Estadual de Campinas, a Universidade Tecnológica Federal do Paraná e a Universidade Federal de Santa Catarina [6]. Neste projeto foi perseguida a inclusão de novas tecnologias aos sistemas ISDB-T e DVB-T, sendo necessário realizar algumas alterações nestes padrões. As principais melhorias estão na utilização de códigos LDPC (*Low Density Parity Check Code*) e do sistema MIMO (*Multiple-Input, Multiple-Output*) com duas antenas transmissoras (esquema de Alamouti). Esse novo sistema foi chamado de MI-SBTVD (Modulação Inovadora para o Sistema Brasileiro de TV Digital), porém, não foi adotado devido a questões burocráticas envolvendo direitos autorais dos padrões modificados.

Assim, o sistema MI-SBTVD não foi adotado, e se optou por escolher um dos padrões já existentes. Vários testes foram feitos em território brasileiro, onde foi atestado que o padrão japonês teve desempenho superior frente a seus principais concorrentes. É

importante citar que pesquisadores brasileiros tiveram participação ativa na versão do ISDB-T adquirida pelo Brasil, denominada ISDB-T<sub>b</sub>, sugerindo algumas adaptações no sistema que foram acatadas pela ARIB. Como exemplos de alterações no projeto temos a codificação de vídeo H.264/AVC (H.264/*Advanced Video Coding*) e o Middleware [7].

Por fim, para “fechar negócio”, a ARIB concordou em não cobrar royalties, referentes ao sistema de transmissão ISDB-T, para fabricação de receptores no Brasil. Essa redução de custo é refletida diretamente no preço do receptor que será adquirido pelo consumidor final.

## 1.1 Motivação e objetivo

Para se obter um bom desempenho por parte do sistema ISDB-T deve-se levar em consideração a diversidade territorial que o Brasil possui. Desta forma, é possível afirmar que a eficiência do sistema está diretamente relacionada com a correta parametrização dos transmissores nas diferentes regiões do país. Deve-se levar em consideração o relevo da região onde se deseja realizar a transmissão e a mídia que se deseja transmitir, sendo ela SDTV (*Standard Definition Television*), EDTV (*Enhanced Definition Television*) ou HDTV (*High-definition Television*).

A eficiência do sistema é dada por uma combinação de várias etapas de tratamento da informação que, por sua vez, possuem diversas variáveis com impacto direto na qualidade final do sinal. A Tabela 1.1 mostra os parâmetros que podem ser ajustados no transmissor.

Parâmetros	Valores
Modos de Operações (subportadoras)	1 (1405); 2 (2809); 3 (5617)
Taxa do Intervalo de Guarda	1/4; 1/8; 1/16; 1/32
Tipos de Modulações	D-QPSK; QPSK; 16-QAM; 64-QAM
Taxa da Codificação Interna	1/2; 2/3; 3/4; 5/6; 7/8
Profundidade de Entrelaçamento	0; 380; 760; 1.520 símbolos OFDM (modo 1) 0; 190; 380; 760 símbolos OFDM (modo 2), 0; 95; 190; 380 símbolos OFDM (modo 3)

Tabela 1.1: Parâmetros de transmissão do sistema ISDB-T

O fato de existirem inúmeras variáveis dificulta o trabalho do operador do transmissor,

que necessita saber qual a melhor combinação dos parâmetros a serem escolhidos, visando obter um desempenho otimizado do sistema. Sabe-se que a configuração correta destes parâmetros depende de variáveis externas dadas pelo canal de comunicação, que por sua vez depende do relevo e do meio em que se deseja transmitir.

Sabendo das dificuldades dos profissionais da área em realizar uma parametrização otimizada dos transmissores, bem como possíveis limitações técnicas dos mesmos, este projeto tem como motivação criar uma ferramenta de consulta visando assessorar tais profissionais, bem como outros profissionais da área de telecomunicações, quando o assunto é ISDB-T<sub>b</sub>. Desta forma, o profissional poderá consultar, no software, quais impactos ocorrerão no sistema para uma determinada configuração de transmissão.

Para atender aos objetivos descritos no parágrafo anterior, o projeto será dividido em duas etapas:

- Criação de um software de consulta;
- Criação de um simulador para gerar os resultados.

O software de consulta será desenvolvido utilizando programação orientada a objetos (Delphi/Pascal) para plataforma Windows. O simulador será desenvolvido inteiramente em Matlab, onde os resultados obtidos através das simulações serão salvos em pastas distintas utilizando um estrutura de arquivos padronizada. Essa padronização é necessária visando facilitar a integração do simulador e do software de consulta. Os resultados das simulações deverão ser exibidos de maneira amigável e clara através de vídeos das constelações, imagens transmitidas e curvas de desempenho.

Esse projeto faz parte do *Programa de Formação de Recursos Humanos em TV Digital* criado pela CAPES. O objetivo do projeto é implantar redes de cooperação acadêmica no país na área de TV Digital, possibilitando a produção de pesquisas científicas e tecnológicas e a formação de recursos humanos pós-graduados no tema. Uma parceria entre a Universidade Federal do Paraná, Universidade Tecnológica Federal do Paraná e Pon-

tífica Universidade Católica do Paraná foi feita e bolsas de estudos foram concedidas para alunos de pós-graduação cujo tema da pesquisa estivesse dentro do cunho do projeto.

## 1.2 Trabalhos relacionados

Alguns trabalhos já foram publicados no sentido de avaliar o desempenho do sistema ISDB-T, porém em condições restritas de parâmetros e/ou perseguindo outros tipos de resultados.

Em [8], é estudado o desempenho do sistema europeu de TV digital com um novo entrelaçador temporal. Neste contexto é feito uma abordagem dos sistemas ISDB-T e DVB-T, contendo um capítulo inteiro dedicado ao estudo dos entrelaçadores. No trabalho apresentado em [9], é feito um estudo dos sistemas ISDB-T e DVB-T com foco na codificação de canal. Neste documento é mostrado um ganho de codificação de canal devido ao uso de códigos BCH concatenados com códigos Turbo e LDPC.

É possível encontrar trabalhos que busquem a implementação de esquemas de múltiplas antenas, como [10], cujo estudo inclui a simulação e análise de sistemas MIMO aplicados na radiodifusão de televisão digital terrestre.

Outro documento que busca a melhoria da qualidade do sistema através de implementação de novos recursos é apresentado em [11]. Naquela tese, é perseguida a otimização de receptores de TV Digital baseados em OFDM, que operam em redes de transmissão em frequência única. Neste contexto, técnicas de estimação de canal, sincronismo e equalização adaptativa usando LMS (*Least Mean Square*) são abordadas e estudadas.

Existem outros trabalhos que documentam a criação de ferramentas de simulação para o sistema ISDB-T, como por exemplo [12], no qual é criada uma ferramenta para simular as etapas de transmissão, não envolvendo etapas de recepção e nem de modelagem de canal. O objetivo de [12] é detalhar a implementação da camada física do sistema de transmissão ISDB-T<sub>b</sub>, cujo sinal gerado pela ferramenta é validado em laboratório. Isso tornou possível criar uma referência mais detalhada que a norma [4], do ponto de vista



do sistema de transmissão do padrão ISDB-T<sub>b</sub>.

Uma referência importante a ser comentada é [13], que tem como principal objetivo a avaliação de desempenho do sistema ISDB-T de TV digital para serviços multimídias. Esta referência torna possível a validação de resultados obtidos através do simulador criado por esse trabalho, pois tal documento traz curvas de desempenho do código convolucional e curvas dos códigos concatenados (Reed Solomon e código convolucional).

Em [14], outra ferramenta com o objetivo de avaliar o desempenho do sistema ISDB-T é criada. Também utilizando o Matlab, tal ferramenta permite fazer simulações e analisar os resultados através de curvas de desempenho. Porém, no caso de canais com multipercursos, um modelo bem simples é utilizado, com apenas dois caminhos. Diferente deste trabalho, o trabalho de [14] utiliza as informações de todas as portadoras de controle do sistema ISDB-T. Este trabalho utiliza apenas as informações das portadoras de controle que são destinadas à equalização de canal. Ainda em [14], o processo de estimação de canal não foi descrito, o que indica uma possível não implementação de técnicas de estimação de canal.

Todos os trabalhos aqui citados não visam criar uma base de dados com os resultados de simulações, tornando a análise de resultados que busca visualizar o ganho/perda de desempenho em virtude da escolha de parâmetros do sistema mais difícil de ser realizada. Uma simulação que apresente resultados satisfatórios para uma determinada combinação de parâmetros pode levar horas para ser concluída. Surge então um grande diferencial deste trabalho, a criação de uma base de dados de resultados juntamente com um software de consulta. Visando facilitar a avaliação dos resultados simulados para diferentes configurações de parâmetros do transmissor.

Outros diferenciais são:

- A criação de um simulador que torne possível variar todos os parâmetros do transmissor ISDB-T<sub>b</sub>, trazidos pela Tabela 1.1. Este objetivo não é perseguido por todos os trabalhos aqui citados.

- Visualização de resultados das simulações através de imagens, vídeos e curvas de desempenho.
- Tornar a interpretação dos resultados simulados possível para usuários menos instruídos por meio do software de consulta. Para isso, o software irá contar com tópicos de ajuda e janelas de consulta a uma base de dados de informações que possa ser atualizada pelos próprios usuários do software, permitindo a troca de informações entre esses usuários.

### 1.3 Estrutura da dissertação

Inicialmente serão revisados alguns conceitos da teoria de comunicações digitais relevantes para o entendimento deste trabalho, juntamente com uma breve introdução ao sistema OFDM, justificando a escolha deste sistema multiportadoras para o padrão ISDB-T. Assim, serão abordadas suas vantagens e desvantagens, do ponto de vista conceitual.

Dando sequência aos capítulos, o sistema ISDB-T<sub>b</sub> será detalhado, onde serão explicadas as diversas etapas de tratamento do sinal que este sistema prevê. Busca-se mostrar o sistema ISDB-T<sub>b</sub> como um todo, porém, dando prioridade às etapas relevantes para este trabalho.

Em seguida, será relatada a metodologia usada para o alcance do objetivo proposto por este trabalho. Neste capítulo, serão abordadas as considerações de implementação feitas, detalhes de estruturas, justificando cada escolha tomada na criação dos *scripts* do simulador.

Dando continuidade, um capítulo com os resultados do simulador será criado visando mostrar os resultados que o simulador pode gerar. Esses resultados podem ser estendidos para todas as combinações de parâmetros desejáveis. Este capítulo terá o objetivo de validar as curvas obtidas utilizando como base a comparação com resultados apresentados na literatura científica consultada. Também será feita uma análise do bloco de estimação de canal utilizando resultados obtidos pelo simulador.

Por fim, busca-se também dar uma visão geral da estrutura de funcionamento do software de visualização de resultados, mostrando recursos que o programa terá. Serão mostradas imagens da interface gráfica e detalhamento de alguns recursos compreendidos pelo mesmo.

## CAPÍTULO 2

---

### Revisão de Conceitos

---

#### 2.1 Introdução

O objetivo deste capítulo é facilitar o entendimento do sub-sistema de transmissão do Sistema Brasileiro de TV Digital, bem como do simulador criado. Esta revisão será baseada na estrutura geral de um sistema de comunicação digital mostrada na Figura 2.1.

O bloco de codificação de fonte é responsável pela compressão da fonte de informação. No caso de sistemas de TV digital, este bloco também se encarrega de multiplexar as diferentes fontes de informação (vídeo, áudio, informação de controle), produzindo um vetor de dados que recebe o nome de pacote de transporte TS (*Transport Stream*).

Na saída do bloco de codificação de fonte, a sequência de dados é entregue para a codificação de canal, responsável por inserir redundância à informação, visando a correção de erros ocorridos no processo de transmissão. A capacidade de correção de erros do código corretor depende da quantidade de redundância de informação que é adicionada.

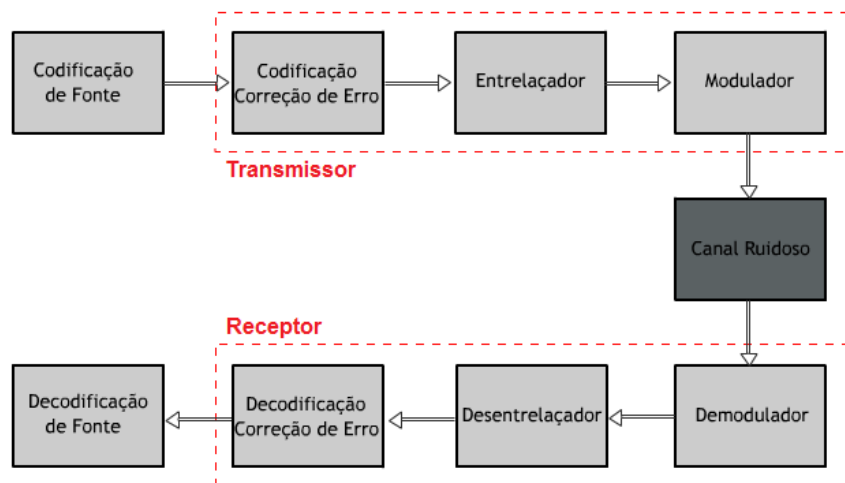


Figura 2.1: Exemplo de um sistema de comunicação digital.

Após a codificação para correção de erros encontramos um bloco de entrelaçamento que tem o objetivo de embaralhar os dados visando aumentar a eficiência do código corretor prevendo a distribuição dos erros entre várias palavras codificadas, trazendo a quantidade de erros para dentro da capacidade de correção do código utilizado.

Após o entrelaçador os dados são modulados por um sistema de portadora simples ou multiportadora. No caso do sistema ISDB-T<sub>b</sub> é usado um sistema de modulação multiportadora. Após modulada, a mensagem é transmitida por um canal ruidoso até chegar ao receptor.

## 2.2 Sistemas multiportadoras

Os sistemas multi-portadoras se caracterizam por transmitir vários símbolos ao mesmo tempo, carregados por portadoras em frequências distintas. Tal técnica difere da utilizada nos sistemas de portadora única, que transmitem os símbolos de forma sequencial, cuja portadora ocupa todo o espectro de frequência disponível. A Figura 2.2 mostra os espectro de frequência de um sistema de portadora única, enquanto a Figura 2.3 mostra o espectro de frequência para um sistema multiportadora.

O fato da portadora única ocupar todo o espectro de frequência ocorre devido a esta

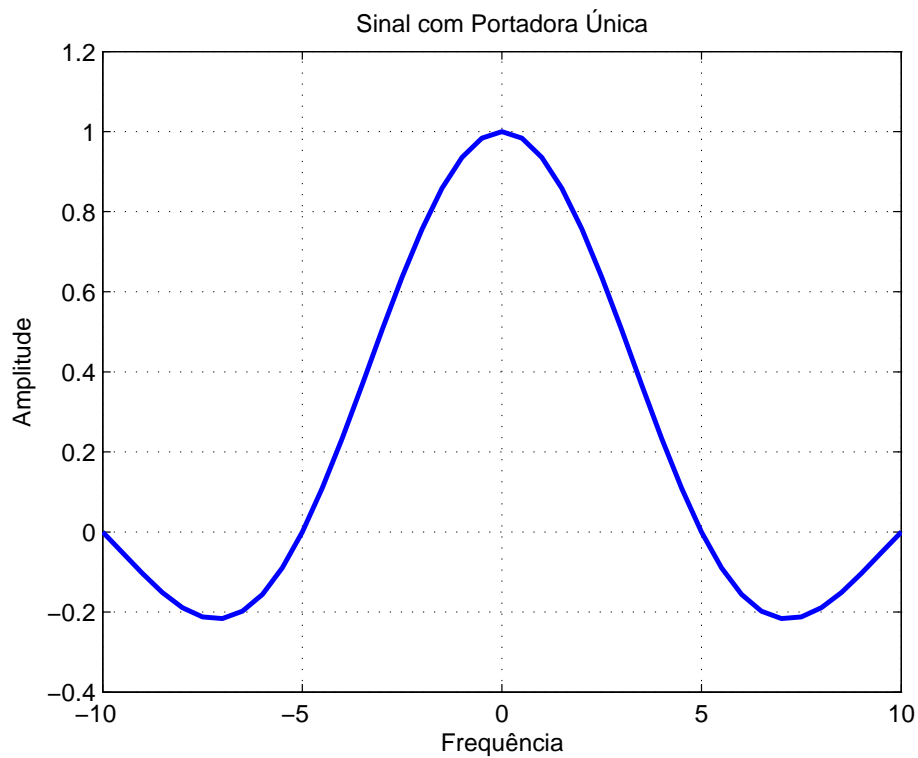


Figura 2.2: Espectro de frequência de um sinal modulado por portadora única.

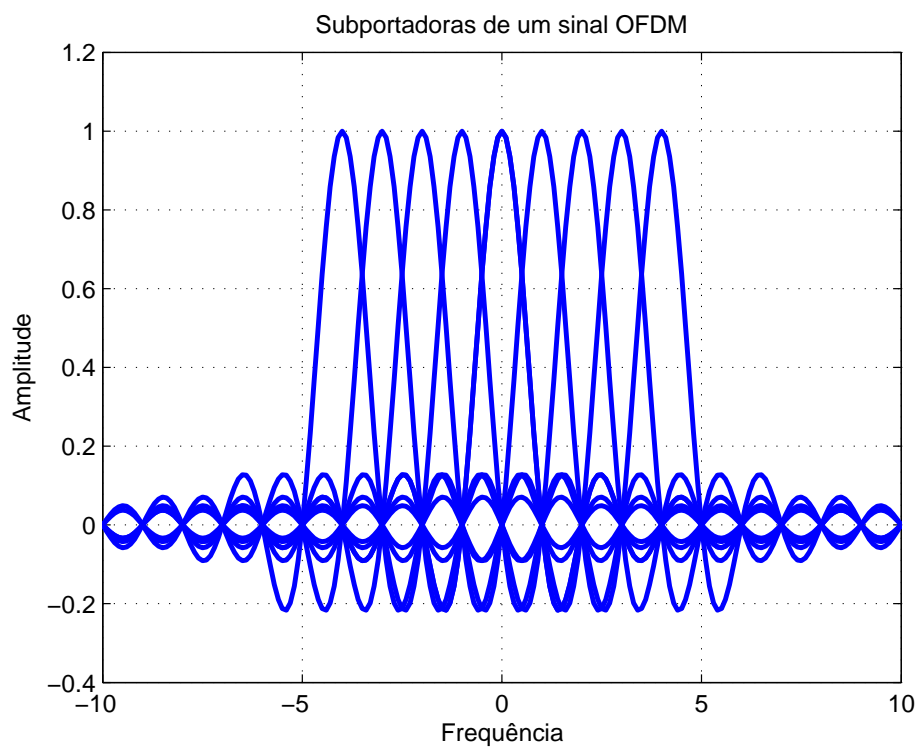


Figura 2.3: Espectro de frequência de um sinal multiportadoras.

portadora ser chaveada muito mais vezes por unidade de tempo que as subportadoras dos sistemas multiportadoras. A quantidade de subportadoras,  $N_c$ , irá indicar o quanto de redução no período de chaveamento será obtido, o que irá reduzir o tamanho dos lóbulos laterais da portadora, permitindo transmitir outras portadoras adjacentes sem que haja interferência entre elas.

A parametrização de um sistema multiportadora não é uma tarefa fácil, pois exige levar em consideração fatores que disputam entre si, por exemplo:

- A largura de banda disponível ( $B_w$ );
- A taxa de transmissão ( $R$ );
- Espalhamento Doppler do canal ( $fd_{max}$ );
- Reflexão de sinal por multi-percursos.

Para preencher as subportadoras, os dados devem ser convertidos de serial para paralelo. A duração do símbolo após a conversão é dada por:

$$T_s = N_c T_d , \quad (2.1)$$

onde  $T_s$  é a duração do símbolo após a conversão serial-paralelo e  $T_d$  é a duração de um símbolo da fonte.

Nota-se que há um aumento na duração do símbolo após essa conversão, e isso justifica a utilização do sistema multiportadora, pois, com o aumento na duração do símbolo, que depende diretamente da quantidade de subportadoras, torna-se o sistema mais robusto contra os multipercursos, porém mais sensível ao efeito Doppler.

O efeito Doppler causa um espalhamento na frequência do sinal, enquanto as interferências por multipercursos acarretam em um espalhamento temporal do sinal. Assim, para tornar o sistema mais robusto contra interferências por efeito Doppler, busca-se diminuir a quantidade de subportadoras, pois, quanto maior a frequência Doppler, menor

o tempo de coerência ( equação 2.11). Assim, busca-se diminuir também o tempo de símbolo para deixá-lo dentro do intervalo de tempo de coerência. Diminuindo a quantidade de subportadoras, ocorre um aumento de espaçamento entre elas, diminuindo as interferências entre portadoras causadas pelo espalhamento de frequência das mesmas.

Para tornar o sistema robusto frente ao multipercurso, busca-se aumentar o número de portadoras, diminuindo o espaçamento entre elas. Com o aumento do número de portadoras,  $T_s$  aumenta, fazendo com que a duração de símbolo seja maior que o tempo de atraso de multipercurso,  $T_{\Delta path}$  abordado na seção 2.4, prevenindo interferência intersimbólica, ISI (*Inter-Symbol Interference*).

### 2.3 O sistema OFDM

Tal técnica é semelhante à multiplexação por divisão de frequência FDM (*Frequency Division Multiplexing*), porém a técnica OFDM utiliza o espectro de forma otimizada.

O princípio da técnica OFDM está em modular os  $N_c$  símbolos com subportadoras a um espaçamento de frequência ( $F_s$ ) dado por:

$$F_s = \frac{1}{T_s} , \quad (2.2)$$

para atingir ortogonalidade entre as subportadoras, presumindo-se uma envoltória retangular do espectro como resultado.

Os símbolos paralelos da fonte, mapeados em sinais complexos por alguma das constelações que serão vistas na Seção 3.7, serão modulados por subportadoras que ocupam posições específicas no espectro de frequências, sendo que a soma de todas estas subportadoras formam um símbolo OFDM. O sinal complexo,  $x(t)$ , que representa um símbolo OFDM é dado por:

$$x(t) = \frac{1}{N_c} \sum_{n=0}^{N_c-1} S_n e^{j2\pi f_n t} , \quad 0 \leq t < T_s , \quad (2.3)$$



onde  $f_n$  representa as posições das subportadoras no espectro de frequência, e pode ser calculada com:

$$f_n = \frac{n}{T_s}, \quad n = 0, \dots, N_c - 1. \quad (2.4)$$

A Figura 2.4 mostra o espectro de frequências de um sinal com 16 subportadoras.

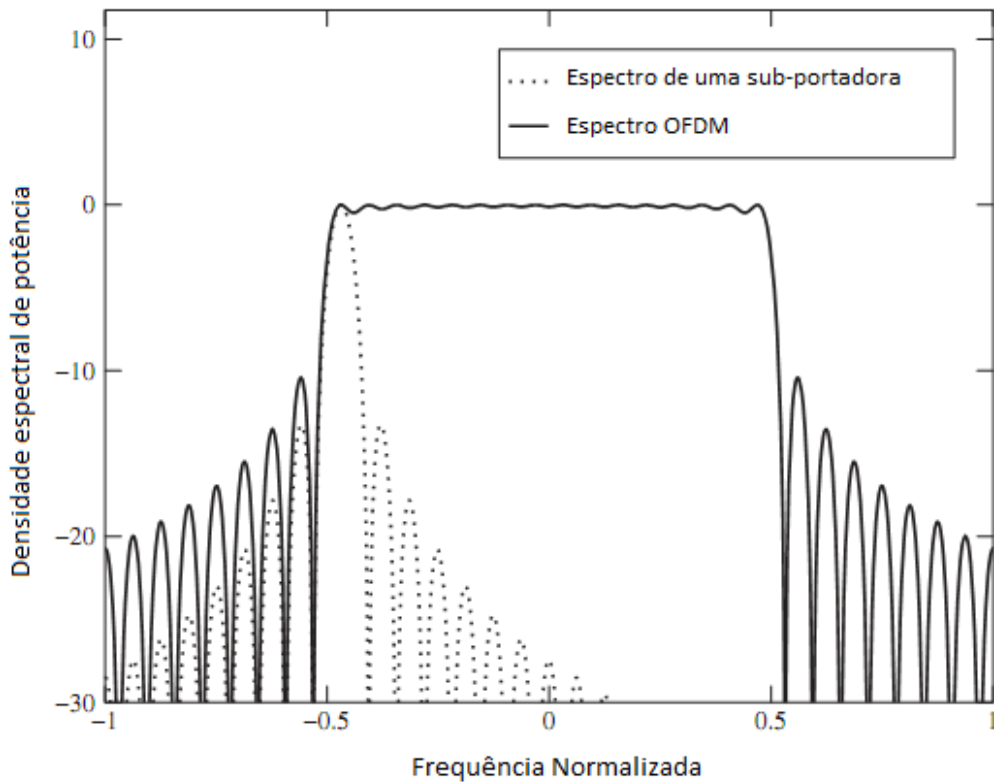


Figura 2.4: Espectro OFDM para 16 subportadoras.

No caso do sistema ISDB-T, a variável  $S_n$  da equação (2.3) representa símbolos complexos pertencentes às constelações de modulação QAM ou  $\pi/4$ -DQPSK.

Um sistema OFDM básico pode ser criado com a utilização de funções multiplicadoras seno e cosseno, como ilustrado na Figura 2.5. Neste sistema, as subportadoras são moduladas variando a frequência angular ( $W_n$ ) das funções seno e cosseno para cada símbolo.

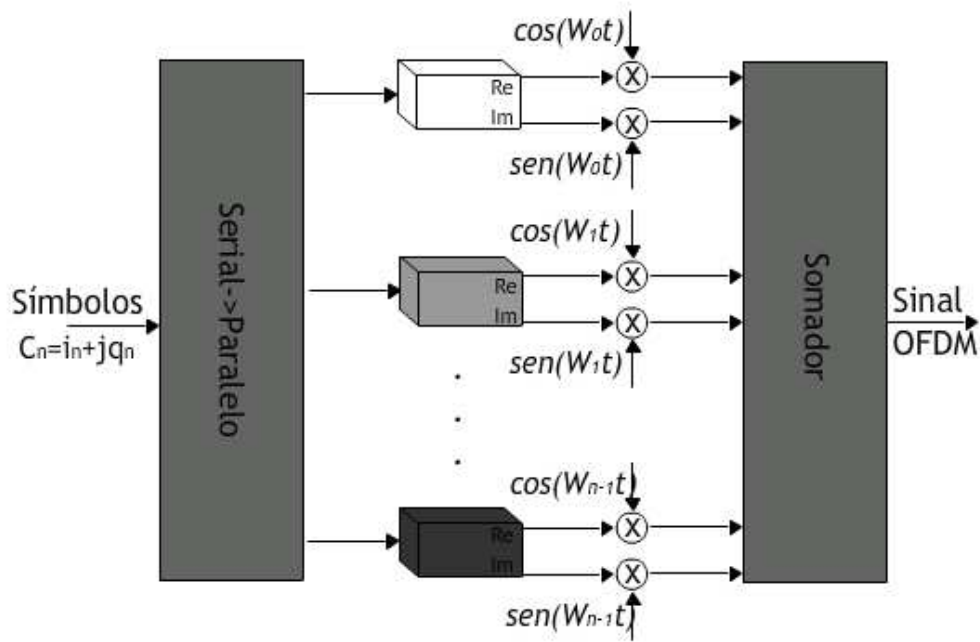


Figura 2.5: Modulador OFDM simples.

### 2.3.1 Discretização

Reescrevendo a equação 2.3 de forma discreta, considerando uma taxa de amostragem de  $1/T_d$ , tem-se [15]:

$$x_v = \frac{1}{N_c} \sum_{n=0}^{N_c-1} S_n e^{j2\pi nv/N_c}, \quad v = 0, \dots, N_c - 1. \quad (2.5)$$

A sequência amostrada  $x_v$ ,  $v = 0, 1, 2, \dots, N_c - 1$ , é a IDFT (*Inverse Discrete Fourier Transform*) dos símbolos na saída do conversor serial-paralelo,  $S_n$ ,  $n = 0, \dots, N_c - 1$ . A saída da IDFT nos retorna um sinal no domínio do tempo, então pode-se considerar que o vetor de símbolos complexos na entrada da IDFT representa o espectro de frequências do símbolo OFDM. A sequência demodulada ( $R_n$ ) é dada por:

$$R_n = \sum_{v=0}^{N_c-1} y_v e^{-j2\pi nv/N_c}, \quad n = 0, \dots, N_c - 1, \quad (2.6)$$

onde  $y_n$  é a sequência recebida passada por um conversor analógico-digital a uma taxa de

amostragem de  $1/T_s$ .

No modulador OFDM básico, mostrado pela Figura 2.5, o aumento do número de subportadoras tornava o sistema inviável de ser utilizado. No sistema que utiliza a IDFT surge outro problema, o tempo computacional. O tempo requerido para realizar a IDFT aumenta exponencialmente com o aumento do número de subportadoras. Para resolver esse problema pode-se utilizar um algoritmo mais eficiente computacionalmente, a Transformada Rápida Inversa de Fourier (IFFT (*Inverse Fast Fourier Transform*)). O diagrama de blocos de um transmissor OFDM que utiliza a IFFT é ilustrado na Figura 2.6.

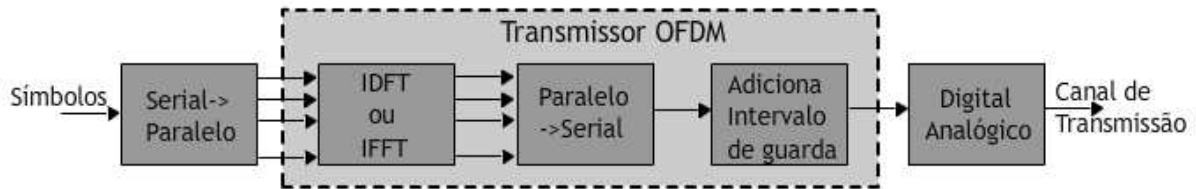


Figura 2.6: Transmissor OFDM com IFFT/IDFT.

### 2.3.2 Intervalo de guarda

De maneira sucinta, quanto maior a quantidade de subportadoras, maior a duração do símbolo OFDM em relação à duração da resposta ao impulso do canal ( $\tau_{max}$ ), o que reduz o risco de que ocorram interferências intersimbólicas [15]. No entanto, para evitar completamente os efeitos da ISI, um intervalo de guarda pode ser inserido entre os símbolos OFDM adjacentes. Como a duração do intervalo de guarda ( $T_g$ ) precisa ser maior que a resposta ao impulso do canal, pode-se escrever:

$$T_g \geq \tau_{max} . \quad (2.7)$$

O intervalo de guarda nada mais é que a extensão cíclica do símbolo OFDM, que visa aumentar a duração deste símbolo. A nova duração do símbolo OFDM ( $T'_s$ ) será então,

$$T'_s = T_g + T_s . \quad (2.8)$$

A quantidade de amostras extras necessárias para prevenir a ISI pode ser calculada fazendo:

$$L_g \geq \left\lceil \frac{\tau_{max} N_c}{T_s} \right\rceil , \quad (2.9)$$

desta forma, a equação 2.5 pode ser reescrita como:

$$x_v = \frac{1}{N_c} \sum_{n=0}^{N_c-1} S_n e^{j2\pi n v / N_c}, \quad v = -L_g, \dots, N_c - 1 . \quad (2.10)$$

Visando manter a ortogonalidade entre as subportadoras, para evitar a interferência ICI, essas amostras extras não podem ter valores nulos, ou seja, o intervalo de guarda não pode ser criado apenas com a ausência de sinal. A Figura 2.7 (Modificada de [16]) ilustra esse problema.

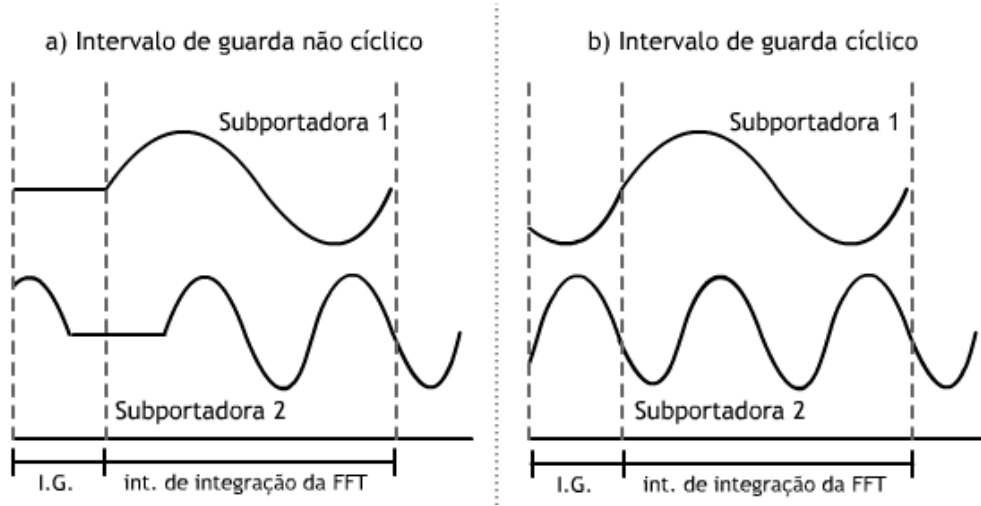


Figura 2.7: Ilustração da necessidade de inserção de prefixo cíclico.

Se a diferença entre o número de ciclos das duas portadoras, dentro do intervalo de integração da FFT, não é inteiro, a segunda portadora irá interferir na primeira portadora e vice-versa. Na parte “a” da Figura 2.7, a subportadora 2 encontra-se retardada devido a algum tipo de interferência, e não possui o prefixo cíclico, possuindo um espaço vazio.

Nessas condições, haverá interferências entre as duas portadoras devido o intervalo de integração da FFT não estar inteiro. Já na parte “b” da figura, o retardo da segunda portadora não irá influenciar no resultado da primeira, pois o prefixo cíclico garante ciclos completos das duas portadoras dentro do intervalo de integração da FFT. Assim, caso o retardo da portadora “2” seja menor que a duração do prefixo cíclico, não ocorrerá ICI [16]. Por fim, a Figura 2.8 ilustra a inserção do intervalo de guarda no início do símbolo OFDM.

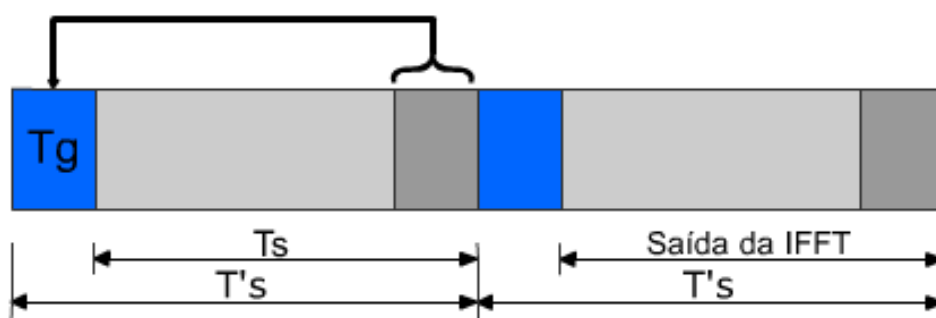


Figura 2.8: Ilustração da inserção do intervalo de guarda ao símbolo OFDM.

### 2.3.3 Sincronismo

Visando a correta demodulação das subportadoras, o receptor deve estar em sincronismo temporal e sincronismo em frequência com o transmissor.

No sincronismo temporal o receptor deve encontrar os limites do símbolo para mitigar os efeitos de ISI e ICI.

No sincronismo em frequência, o receptor deve estimar a variação de frequência das subportadoras, minimizando assim a ICI. É importante citar que qualquer quebra na ortogonalidade das subportadoras resulta em ICI, e para garantir essa ortogonalidade o receptor e o transmissor devem estar em absoluto sincronismo. Erros devido ao ruído de fase dos osciladores do transmissor e do receptor resultam em variação de frequência que causam ICI e ISI no receptor. Uma das desvantagens do sistema OFDM é a grande sensibilidade a falhas de sincronismo, o que pode reduzir o desempenho deste sistema

frente aos sistemas de portadora única.

É possível encontrar na literatura vários métodos de sincronismo, tanto em frequência quanto temporal, como os métodos descritos em [11]:

- Sincronismo temporal:

Métodos baseados na autocorrelação de um símbolo OFDM;

Métodos baseados em correlação cruzada;

Métodos baseados em prefixo cíclico.

- Sincronismo em frequência:

Métodos baseados na autocorrelação de um símbolo OFDM;

Métodos baseados no prefixo cíclico.

Porém, não é de interesse deste trabalho a descrição dos métodos utilizados. Aos leitores interessados em mais detalhes referentes ao sincronismo de sistemas OFDM, consultar [11].

## 2.4 O canal de transmissão

Neste trabalho será estudada a transmissão por propagação em meio livre. Neste tipo de transmissão, os principais efeitos degradantes presentes no canal de comunicação são ilustradas na Figura 2.9.

Os principais tipos de degradações sofridas pelo sinal são divididos em duas classes:

- Desvanecimento em larga escala:

Sombreamento (regiões sem sinal devido a grandes atenuações);

Perdas de propagação (perda de potência de sinal devido à distância entre transmissor e receptor);

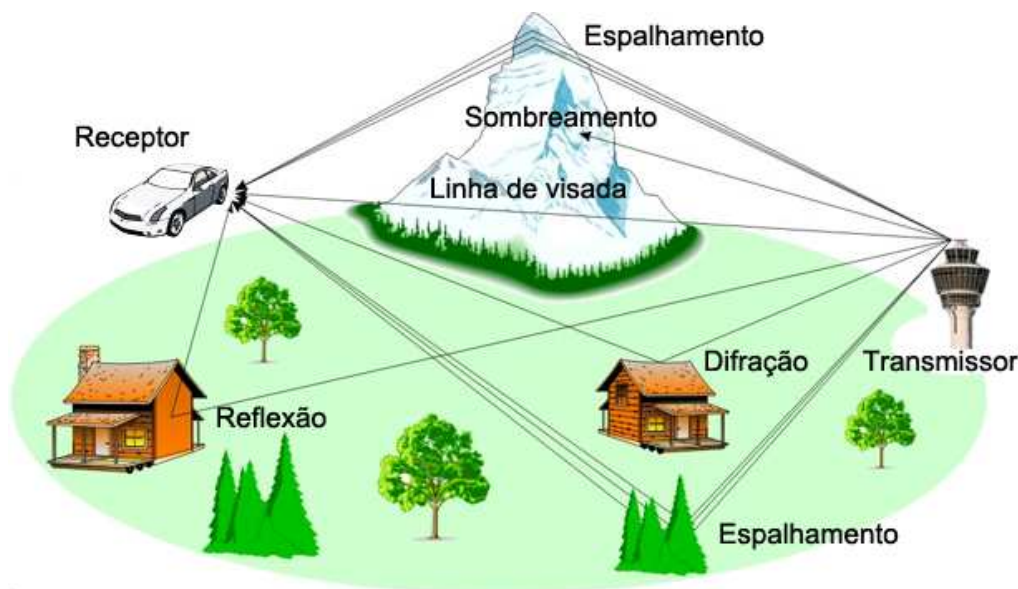


Figura 2.9: Variações temporais de propagação por multipercursos.

- Desvanecimento em pequena escala:

Multipercursos (espalhamento, reflexão e difração);

Efeito Doppler (movimento relativo entre receptor e transmissor).

Os desvanecimentos em larga escala estão associados à localização e posicionamento das antenas transmissoras. Desta forma, os parâmetros do sistema não têm influência significativa nesta classe. São exemplos de desvanecimentos em larga escala a perda de energia do sinal devido à propagação e a forte atenuação do sinal causadas por barreiras naturais ou construções.

O desvanecimento em pequena escala é causado por interferências construtivas e/ou destrutivas das réplicas do sinal geradas pelos multipercursos. Outro tipo de distorção desta classe de desvanecimento é o efeito Doppler.

O problema da propagação por multipercursos são as réplicas do sinal que chegam ao receptor com diferentes fases, devido a diferentes ângulos de chegada e tempos de propagação. A soma destas diferentes réplicas do sinal tem como resultante um sinal distorcido, variando a medida que o usuário se movimenta. Mesmo se o usuário estiver em repouso, aparecerão variações de amplitude e fase do sinal devido a alguns objetos

presentes no meio de transmissão se movimentarem, por exemplo os veículos em centros urbanos.

Se a distância entre o receptor e o transmissor varia no tempo, ocorre uma alteração na frequência do sinal, chamada de deslocamento Doppler, onde cada componente multipercurso sofrerá um desvio diferente. O deslocamento Doppler em um canal com multipercursos ocasiona o espalhamento da largura de banda dos múltiplos sinais dentro de um intervalo dado pela frequência do sinal mais (ou menos) a máxima frequência Doppler.

Buscando o bom entendimento dos tipos de desvanecimentos que estão relacionados ao multipercurso e ao efeito Doppler, deve-se prestar atenção em dois itens:

- Tempo de coerência;
- Banda de coerência.

O tempo de coerência descreve a velocidade com a qual o canal varia no tempo. Se o tempo de coerência é grande, menores são as oscilações da resposta ao impulso do canal, ou seja, se o tempo de coerência for menor que o tempo de transmissão do símbolo, o canal mudará sua resposta durante a transmissão do símbolo, causando distorções ao sinal recebido.

O tempo de coerência ( $T_c$ ) é inversamente proporcional ao espalhamento Doppler e pode ser definido por [17]:

$$T_c \approx \frac{1}{fd_{max}} , \quad (2.11)$$

onde  $fd_{max}$  representa a máxima frequência Doppler do sinal. A mesma análise pode ser feita no domínio da frequência. É a chamada banda ou faixa de coerência. A banda de coerência indica a faixa de frequências em que a resposta do canal se mantém constante.

A banda de coerência ( $B_c$ ) é inversamente proporcional à diferença do tempo de propagação entre o percurso mais longo tomado pelo sinal e o percurso mais curto, ( $T_{\Delta path}$ ). A banda de coerência pode ser representada por [17]:



$$B_c \approx \frac{1}{T_{\Delta path}} . \quad (2.12)$$

Sabendo que o tempo de transmissão de um símbolo é  $T_s$  e a largura de banda é  $B_w$ , é possível analisar os seguintes tipos de desvanecimento:

- Gerado por multipercurso:

Desvanecimento plano ( $B_w \ll B_c$  e  $T_s \gg T_{\Delta path}$ ): A banda do sinal está dentro da banda de coerência, assim toda a faixa de frequência do canal possui ganho constante e resposta de fase linear;

Desvanecimento seletivo em frequência ( $B_w > B_c$  e  $T_s < T_{\Delta path}$ ): A faixa de frequência do sinal transmitido é maior que a faixa de coerência. O canal irá agir de forma parcial no espectro de frequência, causando seletividade em frequências. Assim, o tempo de atraso  $T_d$  será maior que o tempo de símbolo  $T_s$ , resultando em interferência intersimbólica.

- Gerado por espalhamento Doppler:

Desvanecimento rápido ( $T_s > T_c$  e  $B_w < fd_{max}$ ): O tempo de coerência  $T_c$  do canal é menor que a duração do símbolo transmitido  $T_s$ , ou seja, o canal irá sofrer variações durante o período de transmissão de um símbolo. Neste caso, quanto maior for o espalhamento Doppler  $fd_{max}$  em relação à largura de banda do sinal transmitido, maior será a dispersão em frequência;

Desvanecimento lento ( $T_s \ll T_c$  e  $B_w \gg fd_{max}$ ): O tempo de coerência é muito maior que o período de transmissão do símbolo. Assim, o canal se mantém constante durante a transmissão do símbolo, que indica que o espalhamento Doppler é muito menor que a largura de banda do sinal transmitido.

### 2.4.1 Modelagem

A saída do transmissor é entregue ao canal de comunicação, o qual pode ser modelado matematicamente quando se conhecem as suas características. Neste trabalho, o canal de comunicação será modelado em correspondência com a Figura 2.10. O bloco ruído que aparece nesta figura será modelado neste trabalho como sendo ruído branco gaussiano aditivo AWGN (*Additive White Gaussian Noise*) que representará o ruído térmico na entrada do receptor.

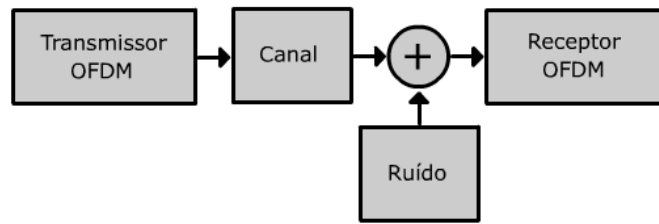


Figura 2.10: Diagrama de blocos de configuração de ruídos do canal.

O efeito de movimento do receptor juntamente com o movimento aleatório de objetos refletores de sinal, pode ser representado pela resposta ao impulso do canal ( $h(t)$ ), que descreve uma relação entre as antenas transmissoras e receptoras. A resposta ao impulso é dada por:

$$h(t) = \sum_{i=0}^{N_{path}-1} A_i \delta(t - \tau_i) , \quad (2.13)$$

onde  $A_i$  e  $\tau_i$  representam, respectivamente, a atenuação e o atraso sofrido pelo sinal no  $i$ -ésimo percurso e  $N_{path}$  corresponde à quantidade de percursos tomados pelo sinal entre o transmissor e receptor.

Contudo, devido à natureza aleatória dos fatores de atenuação e dos atrasos de propagação descritos pela equação 2.13, deve-se adotar um modelo estatístico para descrever seu comportamento.

Os modelos estatísticos de desvanecimento Rayleigh e Rice são utilizados para modelar esse tipo de canal. O desvanecimento Rayleigh é utilizado quando a componente de visada,

ilustrada pela Figura 2.9, não está presente. Já a distribuição Rice leva em consideração a presença desta componente.

## 2.5 Considerações finais

Os conceitos aqui revisados serão vistos no próximo capítulo de forma aplicada ao sistema ISDB-T<sub>b</sub>, juntamente com os valores e parâmetros descritos pela norma [4]. Mais detalhes a respeito dos conceitos introduzidos podem ser encontrados em [18], [15], [19], [17], [20] e [21].

## CAPÍTULO 3

---

### O Sistema ISDB-T

---

#### 3.1 Introdução

A norma brasileira que descreve o sistema ISDB-T de TV digital [4], é baseada na norma escrita pela ARIB [22]. Assim, todo o conteúdo deste capítulo será baseado nestas referências, tornando necessário citar apenas outras referências complementares ao longo do texto. Como não é de interesse reescrever as normas neste documento, será feita uma descrição superficial das etapas de tratamento da informação do transmissor que são relevantes para o entendimento deste trabalho, desprezando detalhes de algumas partes que não serão implementadas ou que não têm importância do ponto de vista do simulador criado, como por exemplo especificações sobre canais UHF (*Ultra High Frequency*) e VHF (*Very High Frequency*) dentre outros.

Uma outra visão que se deve ter em mente é que o sistema simulado é um sistema que não trabalha em tempo real, ou seja, alguns blocos do sistema, como por exemplo o bloco de ajuste de atrasos, não são implementados, o que torna o detalhamento destes blocos

desnecessários. Porém, uma visão superficial será dada para o entendimento do sistema real.

Uma visão macro do transmissor do sistema ISDB-T é dada pela Figura 3.1.

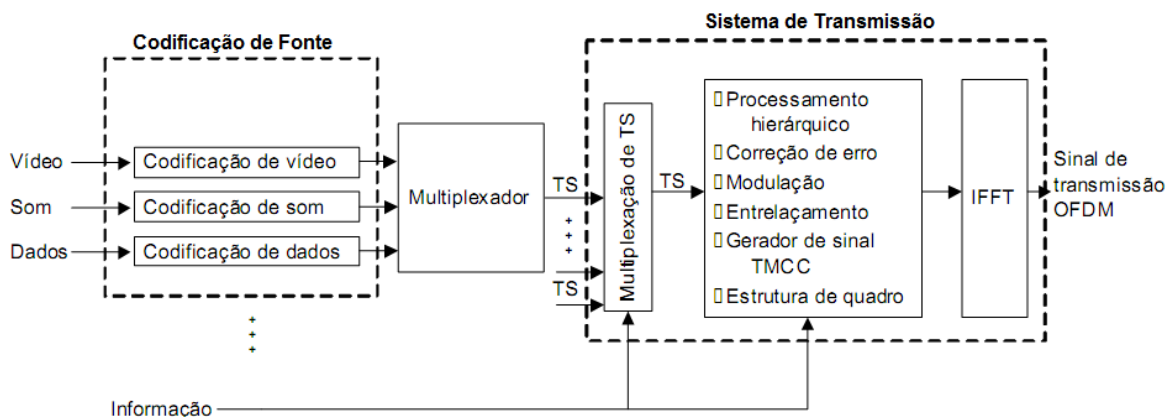


Figura 3.1: Visão macro do sistema de transmissão ISDB-T.

A parte do sistema que se torna relevante de ser abordada, do ponto de vista da escolha de parâmetros, é a etapa de transmissão. Detalhes de implementação e considerações feitas a respeito do sistema ISDB-T, a nível de simulações, bem como os pontos de medição para cálculo de erros de transmissão, serão abordados no Capítulo 4.

Neste capítulo, os itens abordados fazem referência ao sistema de transmissão, pois deve-se entender que não existem normas responsáveis pela construção dos receptores, sendo isso de responsabilidade dos fabricantes. Contudo, a norma prevê estruturas para auxiliar os receptores, como por exemplo as portadoras de controle. Essas são utilizadas pelos receptores para equalização de canal e sincronismo, onde os fabricantes dos receptores poderão utilizar métodos de equalização e estimação que julgarem mais adequados e vantajosos de serem implementados. Detalhes sobre os estimadores de canal e outras peculiaridades a respeito dos esquemas de recepção também serão vistos no Capítulo 4.

Uma visão mais detalhada do sistema de transmissão é apresentada na Figura 3.2.



Figura 3.2: Diagrama de blocos do sistema de transmissão do ISDB-T<sub>b</sub>.

### 3.2 Remultiplexador

A idéia principal da remultiplexação do TS é criar uma estrutura de informação que será utilizada pelo sistema de transmissão. Com esse intuito, um ou mais TS são divididos em estruturas de pacotes, chamadas de TSP. Assim, uma estrutura de  $N$  TSPs é chamada de quadro multiplex.  $N$  deve respeitar a quantidade de informação que irá ser transmitida dentro de um quadro OFDM. Os TSPs são compostos de 204 bytes totais, sendo que os

últimos 16 bytes são nulos. Sua estrutura pode ser vista na Figura 3.3. Futuramente, no bloco de codificação externa, os bytes nulos serão substituídos pela redundância do código RS(204,188). A quantidade de TSPs que irá compor um quadro multiplex pode ser vista na Tabela 6 de [4].

É importante saber que a sequência de TSPs que é criada pelo remultiplexador leva em consideração a divisão hierárquica que irá acontecer em etapas futuras. Assim, os TSPs de cada uma das três camadas são arranjados para formar uma única sequência de pacotes distribuídos de maneira que o circuito de divisão hierárquica possa separá-los entre as camadas desejadas. A Seção 3.9.3 irá detalhar o funcionamento do bloco de divisão hierárquica.

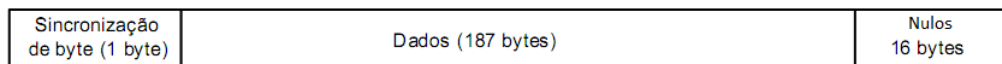


Figura 3.3: Estrutura de um TSP do sistema ISDB-T.

### 3.3 Codificação externa

Modulações como 64-QAM podem sofrer interferências significativas nos dados demodulados devido a presença de ruído no sinal recebido. Assim, se torna necessário a utilização de códigos corretores de erros para baixar a taxa de erro de bits da recepção.

A codificação de canal utilizada no sistema ISDB-T<sub>b</sub> é um esquema de codificação concatenado, que consiste em um código externo e um interno. O código externo é o primeiro a codificar a mensagem e trata-se de um código Reed-Solomon, da classe dos códigos de blocos. As diferentes classes dos códigos corretores de erros é trazida pela Figura 3.4.

Os códigos de blocos codificam  $k$  símbolos inserindo  $n - k$  símbolos de paridade para se obter uma palavra código de  $n$  símbolos. A classe de códigos de Reed Solomon são códigos de bloco lineares, sistemáticos e cíclicos.

Códigos de Reed Solomon, são construídos sobre  $GF(2^m)$ , ver [20], e seu polinômio

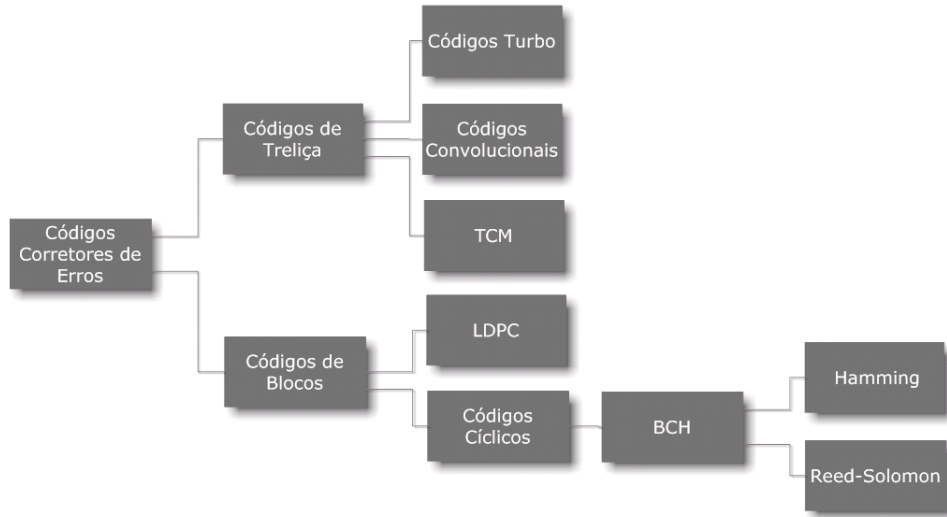


Figura 3.4: Diagrama de bloco das classes dos códigos corretores de erros.

gerador tem a forma:

$$g(x) = (x - \alpha^1) (x - \alpha^2) (x - \alpha^3) \dots (x - \alpha^{2t_{cb}}) , \quad (3.1)$$

onde  $t_{cb}$  representa a capacidade de correção de erros do código e  $\alpha$  é um elemento primitivo de  $\text{GF}(2^m)$ . A equação 3.1 pode ser reescrita como:

$$g(X) = g_0 + g_1X + g_2X^2 + \dots + g_{2t_{cb}-1}X^{2t_{cb}-1} + X^{2t_{cb}} , \quad (3.2)$$

onde  $g_i \in \text{GF}(2^m)$ . Os códigos RS são muito eficientes e fáceis de terem seus parâmetros ajustados, por isso são largamente utilizados na indústria, podendo detectar e corrigir até  $t_{cb}$  erros em qualquer posição da palavra código. Estes códigos são capazes de decodificar símbolos de  $m$  bits em blocos de  $n$  símbolos. No caso do sistema ISDB-T,  $m$  é igual a 8. Os parâmetros dos códigos RS são:

- Tamanho do bloco:  $n = 2^m - 1$  símbolos de  $\text{GF}(2^8)$ ;
- Tamanho da mensagem:  $k$  símbolos;
- Número de símbolos de paridade:  $n - k = 2t_{cb}$ .

Busca-se um código RS com capacidade de codificar um TSP inteiro, ou seja  $k =$



188 bytes. Desta forma, o padrão industrial RS(255,239) é utilizado. Para isso, deve-se adicionar 51 bytes nulos no início do TSP, deixando-o com 239 bytes. Cada bloco codificado pelo código externo possui 255 bytes, com os primeiros 51 bytes sendo nulos. Por fim, removem-se os bytes nulos, resultando em uma palavra código de 204 bytes, que corresponde a um TSP inteiro.

Esse código é construído sobre  $GF(2^8)$ , e a partir do polinômio primitivo:

$$p(x) = x^8 + x^4 + x^3 + x^2 + 1 . \quad (3.3)$$

O polinômio  $g(x)$  usado para gerar o código é calculado como:

$$g(x) = (x - \lambda^0)(x - \lambda^1)(x - \lambda^2) \dots (x - \lambda^{15}) , \quad (3.4)$$

sendo que  $\lambda = 02_{HEX}$ , é o elemento primitivo de  $GF(2^8)$ .

### 3.4 Dispersor de energia

Os blocos de dispersão de energia têm o objetivo de evitar sequências muito grandes de *zeros* ou *uns*. Assim, evita-se a formação de possíveis níveis DC no sinal de transmissão, causando interferência intersimbólica [8]. Essa dispersão de energia acontece de maneira independente entre as camadas hierárquicas do sistema e tem seu circuito gerador ilustrado pela Figura 3.5.

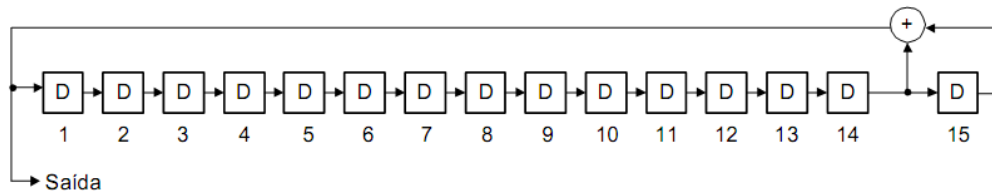


Figura 3.5: Estrutura de dispersão de energia.

A estrutura de dispersão de energia utiliza uma sequência pseudo-aleatória obtida através do polinômio gerador:

$$G(x) = X^{15} + X^{14} + 1. \quad (3.5)$$

Os registradores de deslocamento, mostrados pela Figura 3.5, são carregados inicialmente com uma sequência de bits pré-definida: “100101010000000” [4].

### 3.5 Ajuste de atrasos

O uso de diferentes parâmetros entre as camadas tem como consequência diferentes taxas de transmissão entre elas, acarretando em diferentes atrasos causados pelo bloco de entrelaçamento de bytes, conforme Figura 3.2. Assim deve-se realizar o ajuste de atraso para cada uma das camadas.

No caso do entrelaçamento em nível de bits, localizado na frente do mapeador, também ocorrem atrasos devido aos entrelaçadores serem diferentes para cada tipo de modulação. O tempo de atraso varia de acordo com o número de bits contidos em um símbolo da portadora. Essa diferença no tempo de atraso também deve ser corrigida na entrada do entrelaçador através da adição de um valor de ajuste de atraso de acordo com a Tabela 10 da norma [4].

### 3.6 Codificador interno

O código utilizado na codificação interna é um código convolucional, que é um código binário que faz parte dos códigos de treliça. A Figura 3.6 ilustra um circuito codificador de um código convolucional CC ( $n_{cc} = 2, k_{cc} = 1, m_{cc} = 2$ ), onde  $k_{cc}$  representa o número de entradas,  $n_{cc}$  o número de saídas e  $m_{cc}$  a quantidade de registradores de deslocamento do codificador.

Os bits de informação são inseridos nestes registradores, onde a saída codificada é obtida através de operações de adição módulo-2 entre os dados dos registradores e os dados de entrada. A taxa do código convolucional  $R_{cc}$  é dada por:

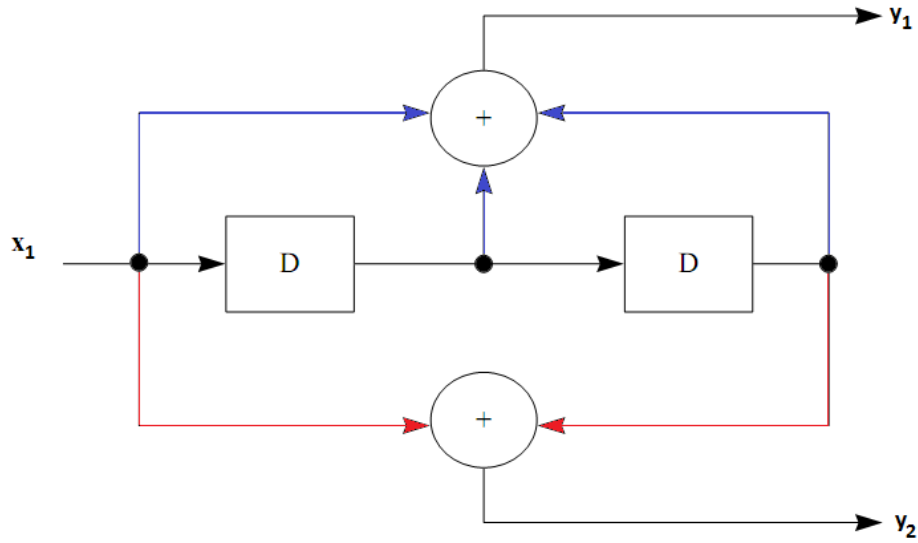


Figura 3.6: Exemplo de um codificador convolucional de  $k = 1, n = 1$  e  $m = 2$ .

$$R_{cc} = \frac{k_{cc}}{n_{cc}}. \quad (3.6)$$

O comprimento de restrição do código (*constraint length*) relaciona o número de bits sobre os quais a saída depende, representado por  $K_{cc}$  e dado por:

$$K_{cc} = m_{cc} + 1, \quad (3.7)$$

para  $k_{cc} = 1$ .

As ligações dos registradores aos somadores módulo-2 podem ser representados por uma sequência geradora. Assim, as ligações registradores-somadores do código, ilustrado pela Figura 3.6, para cada uma das saídas é dado por  $G_1 = [111]$  e  $G_2 = [101]$ .

A taxa do código convolucional pode ser reduzida removendo-se alguns bits da mensagem codificada. Esse procedimento é chamado de puncionamento. O puncionamento de cada saída do circuito codificador tem uma representação polinomial, por exemplo:  $p_{c1} = [1 \ 0]$  e  $p_{c2} = [1 \ 1]$ . Neste exemplo, um bit da primeira saída não será transmitido, representado pelo zero de  $p_{c1}$ . Em suma, a cada quatro bits gerados (dois de cada saída), um será descartado.

A nova taxa de codificação será dada por:

$$R_{cc} = \frac{1/2}{3/4} = 2/3 . \quad (3.8)$$

Desse modo, podem-se obter as taxas de 2/3, 3/4, 5/6 e 7/8.

A Figura 3.7 (Modificada de [8]) mostra a estrutura do codificador e os esquemas de puncionamento utilizados pelo sistema ISDB-T<sub>b</sub>.

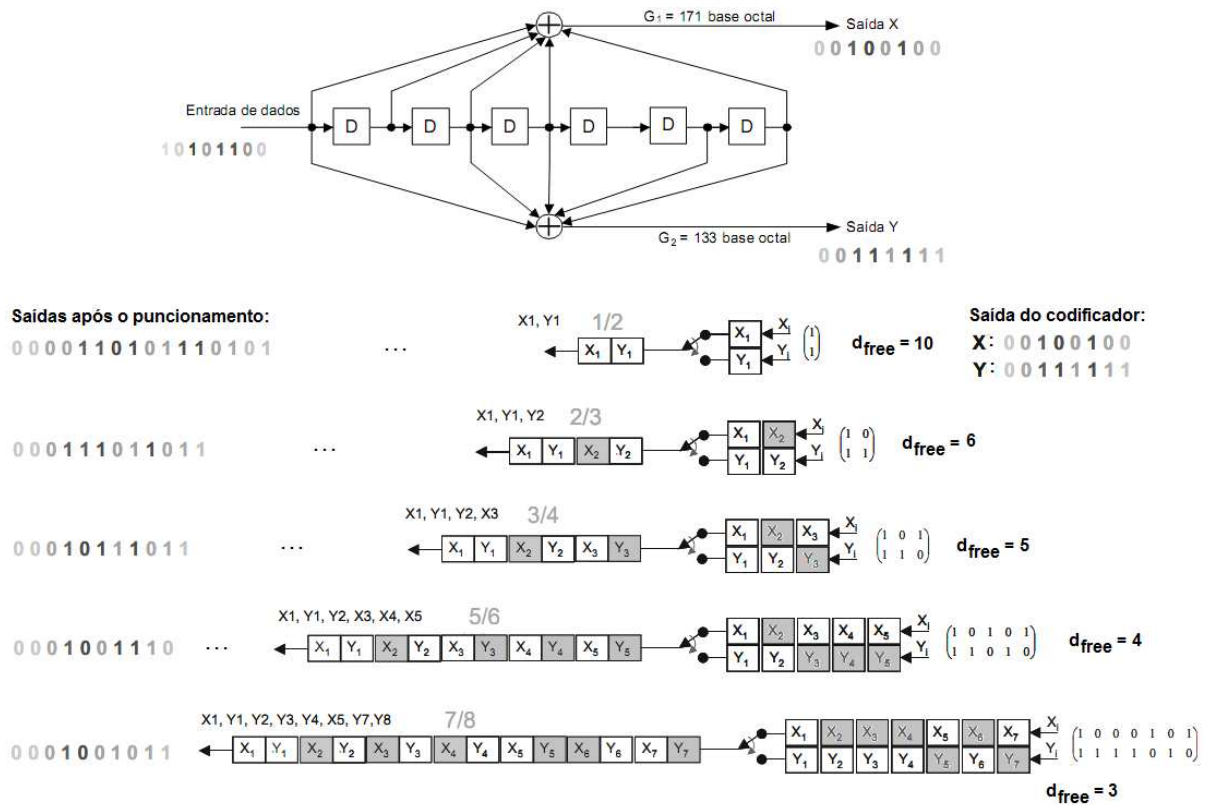


Figura 3.7: Estrutura de codificação e puncionamento do código convolucional utilizado no sistema ISDB-T.

O algoritmo de decodificação mais utilizado junto à codificação convolucional é o algoritmo de Viterbi. Este algoritmo pode operar com dois métodos de decisão, *hard decision* e *soft decision*. No método *hard decision*, os vetores de entrada do decodificador são compostos por “zeros” e “uns”, enquanto no método *soft decision* são utilizados os valores, quantizados, vindos dos demoduladores. O decisor em *hard decision* utiliza a distância de Hamming, enquanto no método *soft decision* é utilizada a distância Euclidiana.

Outro parâmetro importante deste decodificador é a profundidade de decodificação. A profundidade de decodificação é uma janela criada no momento em que o decodificador toma uma decisão, onde os bits desta janela são utilizados para os cálculos que auxiliam o decisor. Este esquema dá à decodificação um ganho devido à utilização de menos memória e menor atraso de decodificação. Em geral é utilizado como profundidade de decodificação um valor cinco vezes maior que  $K_{cc}$ .

### 3.7 Mapeadores

Nesta etapa, os bits são mapeados em símbolos complexos e as constelações utilizadas são:

- $\pi/4$  - DQPSK;
- QPSK;
- 16-QAM;
- 64-QAM.

Uma visão rápida das constelações pode ser obtida analisando a Figura 3.8 (Modificada de [23]), que faz uma relação entre robustez e quantidade de informação transmitida. Porém note que apesar da constelação QPSK ser mais robusta que a DQPSK com rotação de  $\pi/4$ , a última oferece facilidades de sincronismo.

A quantidade de bits por símbolos ( $k_{bits}$ ) destas modulações é dada por:

$$k_{bits} = \log_2(M) , \quad (3.9)$$

onde  $M$  é a quantidade de símbolos distintos que compõe a constelação.

As modulações QAM (*Quadrature Amplitude Modulation*) variam tanto a fase quanto a amplitude do sinal transmitido. Com isso consegue-se obter uma grande quantidade de símbolos distribuídos na constelação.

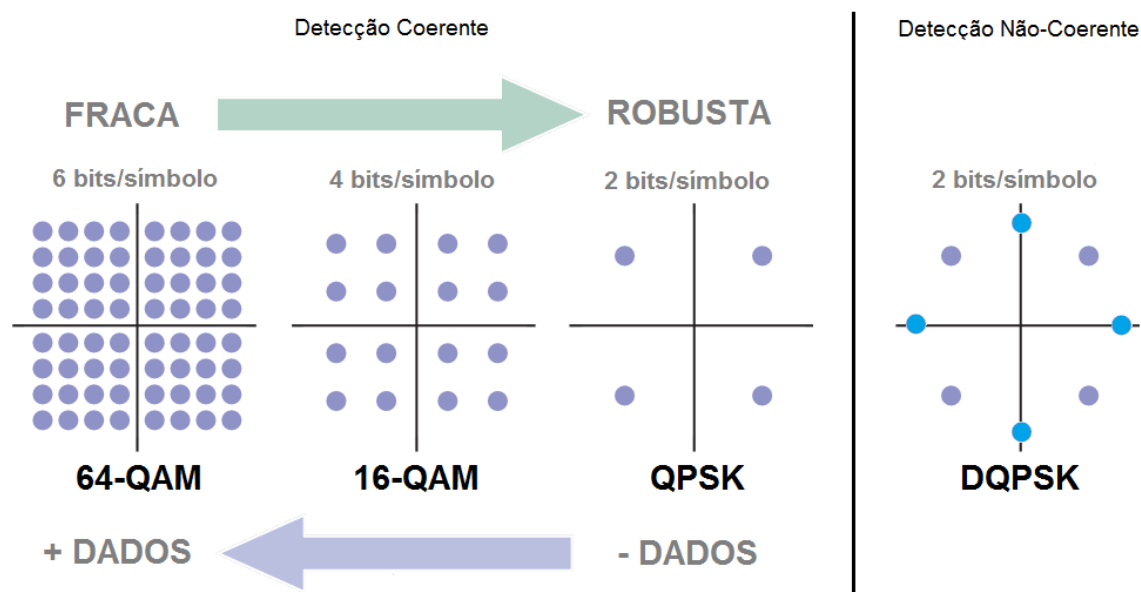


Figura 3.8: Comparativo das constelações utilizadas pelo sistema ISDB-T.

Após o mapeamento, os símbolos gerados são normalizados conforme a Tabela 12 de [4].

O desempenho destas constelações, em canal gaussiano, pode ser visto analisando a Figura 3.9.

### 3.7.1 Detecção coerente e não-coerente

Das quatro modulações utilizadas pelo sistema ISDB-T, uma utiliza detecção não-coerente. Pode-se classificar as modulações de acordo com a estratégia de detecção utilizada na demodulação, são elas:

- Detecção coerente: Utilizada em receptores que são capazes de estar em sincronismo, tanto em frequência quanto em fase, com o oscilador do transmissor, que fornece a portadora utilizada para modular o sinal de mensagem (QPSK, 16-QAM e 64-QAM).
- Detecção não-coerente: O receptor não tem conhecimento da fase da portadora do transmissor (DQPSK).

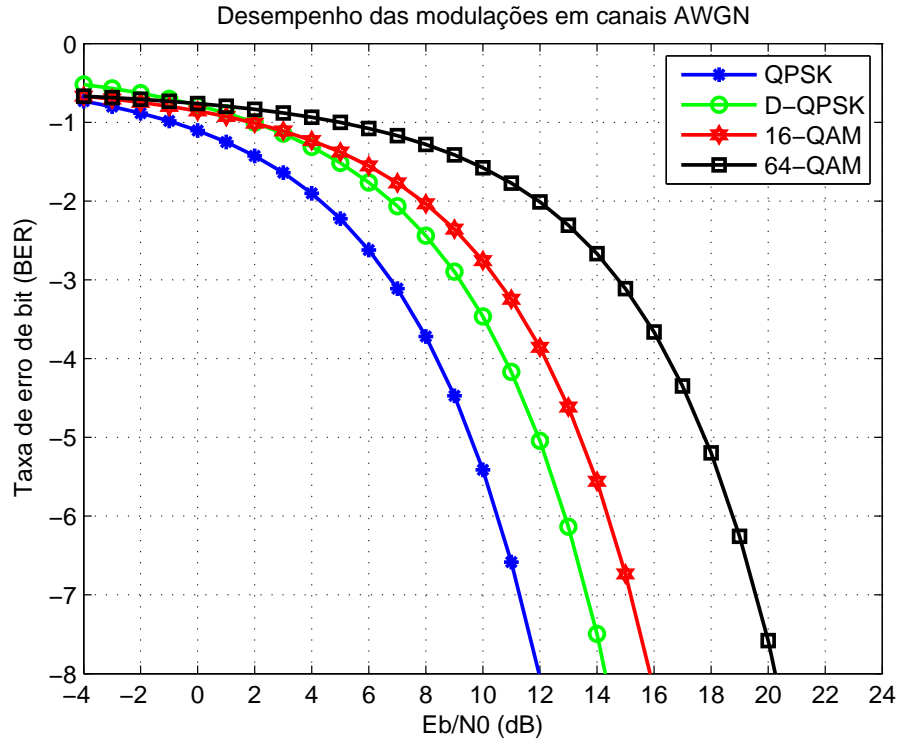


Figura 3.9: Desempenho das modulações do sistema ISDB-T em canais AWGN.

Uma modulação que utiliza detecção não coerente é a modulação DQPSK (*Differential Quadrature Phase Shift Keying*). Nesta modulação, cada símbolo é representado por uma “variação de fase”, diferente da QPSK, cujos símbolos são representados por um ângulo de fase fixo.

A principal vantagem da detecção da modulação DQPSK está na simplicidade do receptor, que não necessita de uma referência de fase para realizar a demodulação, tornando-o mais barato e simples.

A Figura 3.9 mostra a curva de desempenho da modulação DQPSK. Nota-se que o desempenho da QPSK diferencial é um pouco inferior em comparação com a QPSK coerente. Essa queda de desempenho pode ser entendida analisando uma situação hipotética, onde o receptor recebeu um símbolo corrompido com um ruído qualquer. Esse símbolo corrompido será armazenado pelo receptor para utilização na decodificação do próximo símbolo. Se o próximo símbolo vier correto, a diferença entre os dois símbolos vai ser uma variação de fase que pode representar outro símbolo, que por sua vez, vai ser comparado

a outro símbolo, propagando o erro.

### 3.8 Estratégias de decisão dos demoduladores

Nos receptores, as saídas digitais dos demoduladores podem ser obtidas fazendo uso de dois métodos de decisão distintos:

- Decisão Suave (*Soft Decision*);
- Decisão Abrupta (*Hard Decision*).

Na saída do método de decisão abrupta, o sinal é composto por zeros e uns, pois o decodificador leva em consideração o símbolo mais próximo do símbolo recebido para tomar sua decisão.

Já o método de decisão suave entrega, na saída do decisor, valores inteiros quantizados que representam a probabilidade do bit ser “1” ou “0”. Quanto maior for o número inteiro mais certeza o decisor tem de que esse bit é zero. Quanto menor for o número, podendo possuir valores negativos, maior a certeza de que o bit não é um zero, ou seja, maior a certeza de que o bit é “1”.

Existem vários modelos matemáticos utilizados para estimação suave: método dos momentos, método dos mínimos quadrados e o método da máxima verossimilhança. O último método será utilizado por este trabalho.

Alguns algoritmos de máxima verossimilhança já implementados pelo Matlab são:

- Algoritmo de máxima verossimilhança logarítmica exata (*Exact Log-likelihood Ratio Algorithm*);
- Algoritmo de máxima verossimilhança logarítmica aproximada (*Approximate Log-Likelihood Ratio Algorithm*).



O LLR (*Log-Likelihood Ratio*) exato, chamado apenas de LLR, é o logaritmo da probabilidade do bit transmitido ser zero. O LLR para um bit  $b$  qualquer é dado por:

$$LLR(b) = \log \left( \frac{Pr(b = 0 | r = (x, y))}{Pr(b = 1 | r = (x, y))} \right) . \quad (3.10)$$

Assumindo que todos os símbolos possuem a mesma probabilidade de serem transmitidos, o  $LLR(b)$  em um canal AWGN, para uma constelação de dois símbolos é dado por:

$$LLR(b) = \log \left( \frac{\sum_{s \in S_0} e^{-\frac{1}{\sigma^2}((x-s_x)^2 + (y-s_y)^2)}}{\sum_{s \in S_1} e^{-\frac{1}{\sigma^2}((x-s_x)^2 + (y-s_y)^2)}} \right) , \quad (3.11)$$

onde:

- $r$  Sinal recebido em coordenadas  $(x, y)$ ;
- $b$  Bit transmitido;
- $S_0$  Símbolo ideal ou ponto da constelação com bit 0, na posição do bit recebido;
- $S_1$  Símbolo ideal ou ponto da constelação com bit 1, na posição do bit recebido;
- $s_x$  Coordenada de fase do símbolo ideal ou ponto ideal da constelação;
- $s_y$  Coordenada de quadratura do símbolo ideal ou ponto ideal da constelação;
- $\sigma^2$  Variância do ruído em banda base;
- $\sigma_x^2$  Variância do ruído da componente em fase;
- $\sigma_y^2$  Variância do ruído da componente em quadratura.

No LLR aproximado, o cálculo da probabilidade é feito levando em consideração apenas o ponto da constelação mais próximo do sinal recebido, ao contrário do que é feito no LLR, que são levados em conta todos os pontos da constelação. A equação do LLR aproximado para um bit qualquer ( $LLR_{ap}(b)$ ) é dada por:

$$LLR_{ap}(b) = -\frac{1}{\sigma^2} \left( \min_{s \in S_0} ((x - s_x)^2 + (y - s_y)^2) - \min_{s \in S_1} ((x - s_x)^2 + (y - s_y)^2) \right) . \quad (3.12)$$

O método LLR aproximado foi utilizado no simulador criado por este trabalho. Esse método já encontra-se implementado em Matlab e apresenta desempenho satisfatório

quando trabalhado em conjunto com o decodificador do código convolucional (decodificador de Viterbi), que utiliza as saídas quantizadas para tomada de decisões.

## 3.9 O quadro OFDM

### 3.9.1 Modos de operação

Variar a quantidade de subportadoras por símbolo OFDM pode ser interessante do ponto de vista de flexibilidade do sistema perante a interferências do canal de transmissão. Como já discutido no Capítulo 2, pode-se tornar o sistema mais robusto a interferências causadas por multipercursos aumentando o número de subportadoras, o que irá torná-lo mais frágil a interferências por efeito Doppler. Por essa razão o sistema prevê três modos de operação que torna possível o operador escolher qual a melhor configuração para a região que se deseja transmitir. As quantidades de subportadoras por símbolo OFDM utilizadas são:

- Modo 1: 1405 subportadoras;
- Modo 2: 2809 subportadoras;
- Modo 3: 5617 subportadoras.

Essa quantidade de subportadoras são totais, ou seja, portadoras de dados mais portadoras de controle.

### 3.9.2 Estrutura do quadro OFDM

Um quadro OFDM é composto por 204 símbolos OFDM, que pode ser dividido em 13 segmentos no domínio da frequência, como mostra a Figura 3.10.

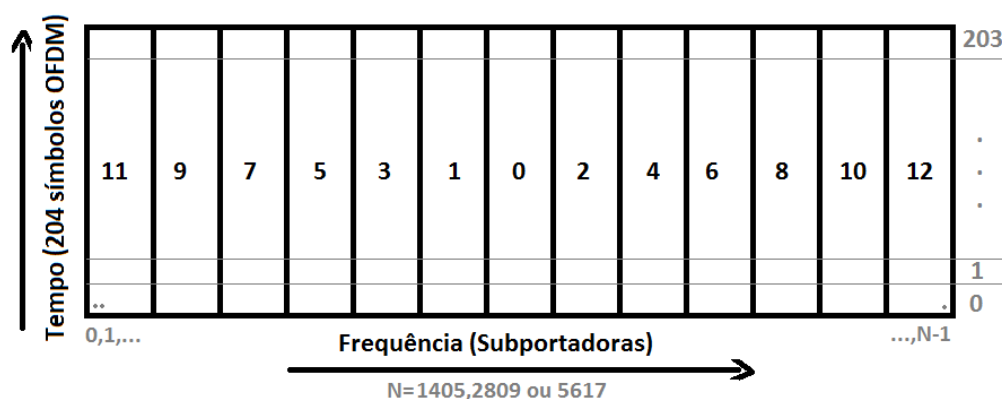


Figura 3.10: Estrutura de um quadro OFDM.

### 3.9.3 Divisão hierárquica

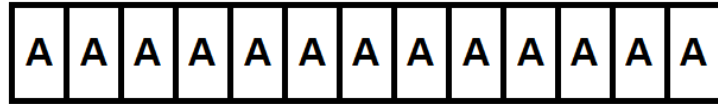
Como pode ser visto na Figura 3.2, o sistema de transmissão prevê três camadas distintas de transmissão, onde cada camada possui taxas de codificação e modulações diferentes e independentes entre si. Pode-se escolher qualquer distribuição de segmentos entre as camadas.

Segundo [19], exemplos de combinações entre número de segmentos e quantidade de camadas são dados pela Tabela 3.1.

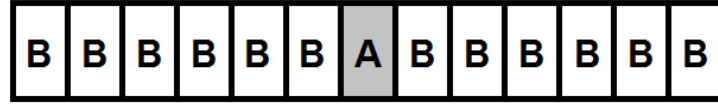
Exemplos	Camada A	Camada B	Camada C
1	13	0	0
2	1	12	0
3	1	3	9

Tabela 3.1: Exemplos de distribuição de segmento entre as camadas hierárquicas.

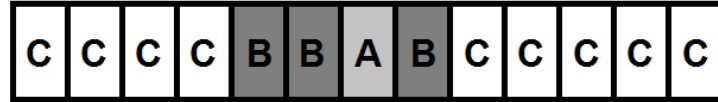
No exemplo 1 é usado apenas uma camada hierárquica, assim os 13 segmentos são usados para transmissão em HDTV ou múltiplas transmissões SDTV. Na segunda opção é utilizado o segmento central para recepção parcial e os demais segmentos para HDTV ou múltiplos SDTV. Por fim, a terceira opção mostra 1 segmento para recepção parcial, 3 segmentos para recepção móvel em SDTV, e nove segmentos para HDTV. A posição das diferentes camadas no espectro de frequências do quadro OFDM é ilustrado pela Figura 3.11.



(a) Distribuição de segmentos para a configuração 1.



(b) Distribuição de segmentos para a configuração 2.



(c) Distribuição de segmentos para a configuração 3.

Figura 3.11: Ilustração do espectro de frequências dos exemplos de distribuição de segmentos às camadas hierárquicas. (a) Configuração 1, (b) Configuração 2 e (c) Configuração 3.

### 3.9.4 Sinais pilotos

Um aspecto interessante para o entendimento deste trabalho é a função de cada um dos quatro tipos de sinais pilotos que o sistema possui. Para se obter um detalhamento mais aprofundado de cada tipo de sinal piloto deve-se consultar a norma [4]. Têm-se como sinais piloto:

- Piloto espalhado, SP (*Scattered Pilot*);
- Piloto contínuo, CP (*Continual Pilot*);
- Controle e configuração de transmissão e de multiplexação, TMCC (*Transmission and Multiplexing Configuration Control*);
- Canal auxiliar, AC (*Auxiliary Channel*).

As portadoras piloto espalhadas (*scattered pilot*) têm seu valor gerado com o auxílio de um circuito gerador de sequências pseudo-aleatórias mostrado na Figura 3.12, que tem o seguinte polinômio gerador:

$$g_{sp}(x) = X^{11} + X^9 + 1, \quad (3.13)$$

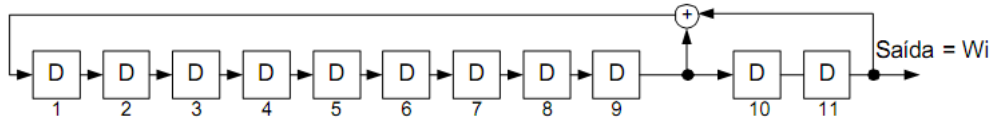


Figura 3.12: Circuito de geração de dados das SPs.

onde  $W_i$  corresponde a  $i$ -ésima portadora do segmento OFDM.

Os valores iniciais dos registradores de deslocamento do circuito de geração apresentado na Figura 3.12 variam para cada segmento e seguem valores da Tabela 23 da norma brasileira [4]. A saída do circuito de geração é mapeada com  $-4/3 + j0$  para “1” e  $4/3 + j0$  para “0”.

As portadoras chamadas de pilotos contínuas também devem ser moduladas com modulação BPSK, levar em consideração a posição da portadora no segmento e também usar o valor de  $W_i$ . Outra consideração a ser feita é que o ângulo de fase da CP com relação à posição da portadora deve ser constante em todo o símbolo.

As CPs possuem a mesma função que as portadoras SPs, a diferença é que as CPs são utilizadas em modulações diferenciais e as SPs em modulações síncronas. A única exceção é que à direita de cada quadro OFDM é inserida uma portadora CP.

O sinal TMCC utiliza modulação DBPSK e tem um código corretor de erro próprio, o código de diferença cíclica (273,191) encurtado para (184,102). Composto por 204 bits, o sinal TMCC carrega a informação das taxas dos códigos corretores de erros para cada uma das camadas, tipos de modulações, profundidade de entrelaçamento, sincronismo entre outros.

A Figura 3.13 (Modificada de [23]) ilustra a distribuição destas portadoras de controle para modulações síncronas e diferenciais para um segmento OFDM criado em modo 1.

Por fim, as portadoras de canais auxiliares transportam sinais adicionais para controle de sinal de transmissão, também utilizando modulação DBPSK. São previstos pela norma dois canais auxiliares (AC1 e AC2) para modulação diferencial e apenas um canal (AC1) para modulação síncrona. Para garantir diversidade de aplicações, somente mo-

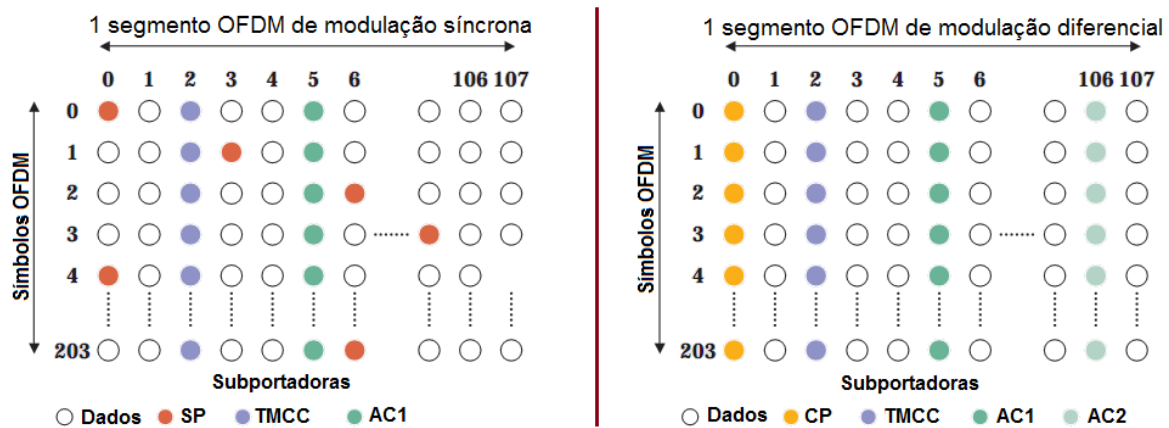


Figura 3.13: Exemplo de distribuição das portadoras de controle em um segmento OFDM modo 1.

dulação DBPSK deve ser aplicada ao sinal. Na ausência de informações auxiliares, serão transmitidos apenas bits “uns” para preenchimento destas portadoras.

É importante saber que a quantidade destas portadoras varia com o tipo de modulação utilizada no segmento e com o modo de operação. A Tabela 3.2 mostra a distribuição destas portadoras.

Tipo de portadoras	Modo 1	Modo 2	Modo 3
Total	$108 \times S_t + 1 = 1405$	$216 \times S_t + 1 = 2809$	$432 \times S_t + 1 = 5617$
Dados	$96 \times S_t = 1248$	$192 \times S_t = 2496$	$384 \times S_t = 4992$
SP	$9 \times S_s$	$18 \times S_s$	$36 \times S_s$
CP	$S_d + 1$	$S_d + 1$	$S_d + 1$
TMCC	$S_s + 5 \times S_d$	$2 \times S_s + 10 \times S_d$	$4 \times S_s + 20 \times S_d$
AC1	$2 \times S_t = 26$	$4 \times S_t = 52$	$8 \times S_t = 104$
AC2	$4 \times S_d$	$9 \times S_d$	$19 \times S_d$

Tabela 3.2: Quantidade de portadoras de controle para um símbolo OFDM.

onde  $S_s$  é o número de segmentos de modulação síncrona,  $S_d$  é o número de segmentos de modulação diferencial e  $S_t$  é o número total de segmentos ( $S_s + S_d$ ).

As posições das portadoras CP, AC e TMCC variam com a numeração do segmento em que estão contidas. No caso de segmentos com modulações diferenciais, as posições das portadoras CP, AC e TMCC nos modos 1, 2 e 3 são dadas pelas Tabelas 17, 18 e 19 da norma [4]. Para modulações síncronas, as posições das portadoras AC e TMCC dentro dos segmentos são dadas pelas Tabelas 20, 21 e 22 de [4].

### 3.10 Entrelaçadores

Os entrelaçadores têm o objetivo de embaralhar a informação de um vetor, como já dito em 2.1, aumentando o desempenho do esquema de correção de erros. Na literatura, pode-se encontrar dois tipos clássicos de entrelaçadores [21]:

- Entrelaçadores de blocos;
- Entrelaçadores convolucionais.

Nos entrelaçadores de blocos, os dados são arranjados em forma de matriz, com o objetivo de embaralhá-los, lendo-os no sentido perpendicular ao sentido que foi distribuído. Assim, se os dados são arranjados no sentido das linhas, o entrelaçador irá lê-los no sentido das colunas, gerando uma sequência diferente na saída. O entrelaçador de blocos pode ser representado por uma matriz  $M_{l,c}$ , sendo o período do entrelaçador  $T_{ent} = lxc$ . Um diagrama de funcionamento deste tipo de entrelaçador é trazido pela Figura 3.14 (Modificada de [8]).

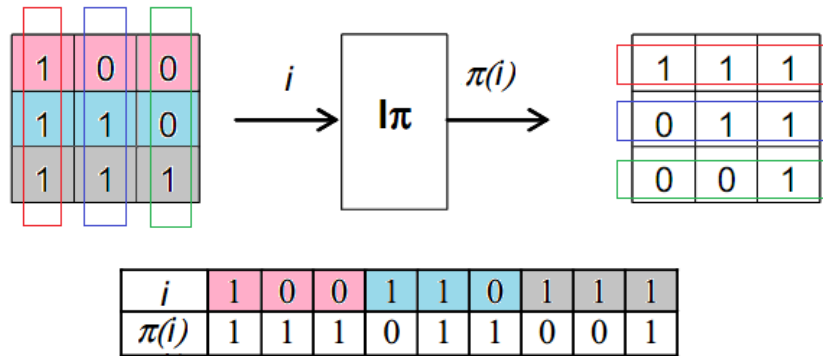


Figura 3.14: Diagrama ilustrativo de um entrelaçador de bloco.

Já os entrelaçadores convolucionais utilizam registradores de deslocamento para atrasar os bytes ou bits, como mostra a Figura 3.15 (Modificada de [8]), que ilustra um entrelaçador convolucional de período  $T_{ent} = 3$  e  $\Delta = 1$ .

Alguns parâmetros de escolha que devem ser levados em consideração na hora de escolher um entrelaçador são [8]:

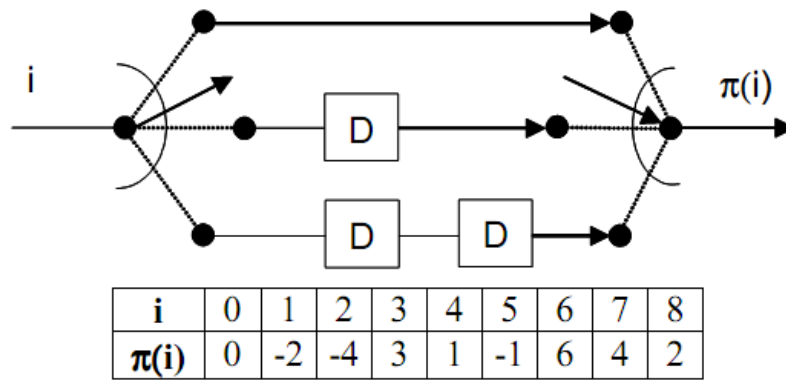


Figura 3.15: Ilustração do entrelaçador convolucional de  $T_{ent} = 3$  e  $\Delta = 1$ .

- Atraso e latência: Necessário serem calculados para um sistema que trabalhe em tempo real, para que tal atraso possa ser corrigido;
- Causalidade: Um entrelaçador é causal quando em um instante  $i$ , sua saída  $\pi(i)$  depende apenas de sua entrada atual ou anterior;
- Memória: Se refere à memória física do entrelaçador, que deve ser calculada para implementação em hardware;
- Fator de Espalhamento: Dado por duas medidas, onde uma indica a distância em que o símbolo “saltou” em relação ao símbolo anterior e a outra, indica a distância mais curta entre os símbolos adjacentes.

No sistema ISDB- $T_b$ , existem quatro blocos de entrelaçamentos distintos. De forma geral, todos têm o mesmo objetivo, aumentar a eficiência dos códigos corretores, evitando sequências de erros nos dados através da aleatorização dos mesmos.

### 3.10.1 Bloco de entrelaçamento de byte

Esse entrelaçador é o primeiro bloco de entrelaçamento que aparece no sistema e sua saída é entregue ao codificador convolucional. Esse entrelaçador é do tipo convolucional e sua estrutura de entrelaçamento pode ser vista na Figura 3.16 (Modificada de [4]).



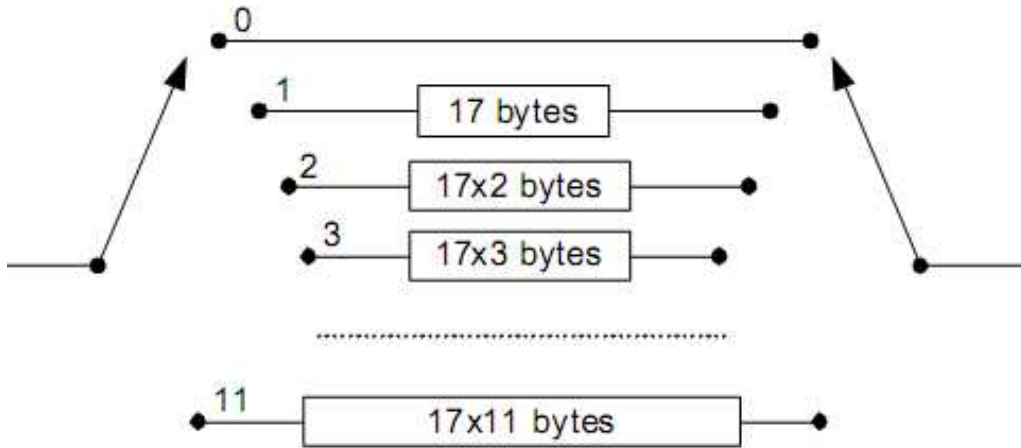


Figura 3.16: Estruturas de entrelaçamento de bytes do sistema ISDB-T.

Esse entrelaçador possui registradores de deslocamento de comprimento igual a 17 bytes, do tipo FIFO (*First In First Out*), onde o primeiro byte a entrar é o primeiro byte a sair. Essa estrutura de entrelaçamento trabalha com 204 bytes, ou seja, um TSP. Portanto, a quantidade de caminhos paralelos é  $204/17 = 12$ .

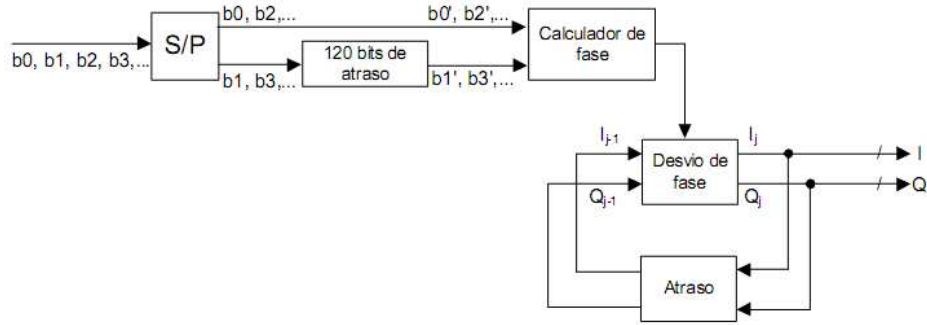
O atraso de cada caminho pode ser calculado como:

$$D_c = 17c \quad , c = 0, \dots, 11 \quad , \quad (3.14)$$

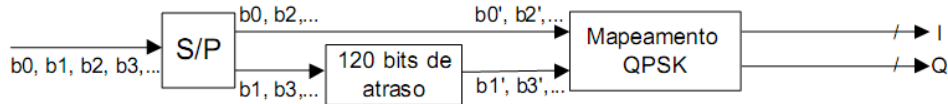
onde  $c$  representa o número do caminho paralelo do entrelaçador. O primeiro caminho não tem atraso ( $D_0 = 0$ ). Isso garante que o primeiro byte do TSP, byte de sincronismo, não será deslocado. A saída entrelaçada é obtida comutando os caminhos de maneira sequencial e cíclica.

### 3.10.2 Bloco de entrelaçamento de bit

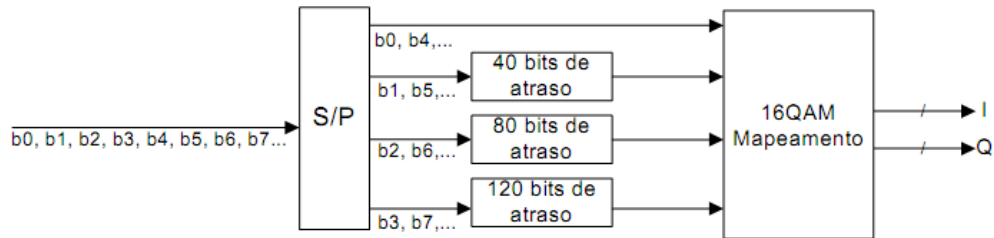
Esse é o segundo bloco de entrelaçamento a aparecer no sistema de transmissão, localizada antes do mapeamento dos bits. Cada tipo de mapeamento (DQPSK, QPSK, 16-QAM e 64-QAM) possui uma configuração distinta de entrelaçamento de bits. A Figura 3.17 (Modificada de [4]) ilustra essas estruturas de entrelaçamento.



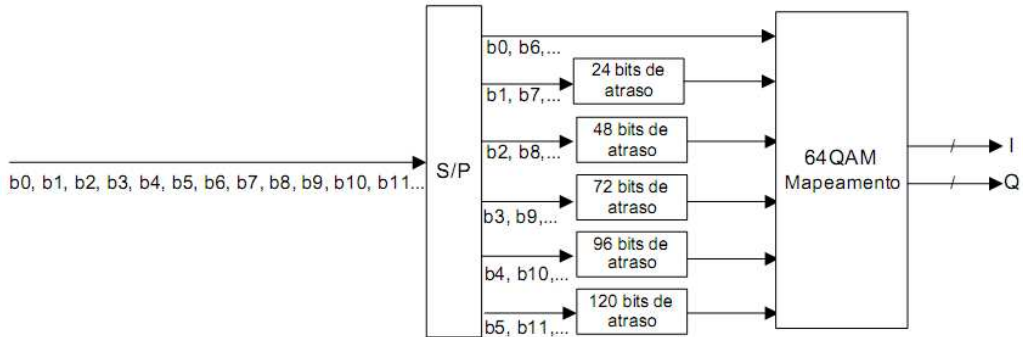
(a) Estrutura de entrelaçamento de bits para modulação DQPSK.



(b) Estrutura de entrelaçamento de bits para modulação QPSK.



(c) Estrutura de entrelaçamento de bits para modulação 16-QAM.



(d) Estrutura de entrelaçamento de bits para modulação 64-QAM.

Figura 3.17: Estruturas de entrelaçamento de bits do sistema ISDB-T. (a) Modulação DQPSK, (b) Modulação QPSK, (c) Modulação 16-QAM e (d) Modulação 64-QAM.

### 3.10.3 Bloco de entrelaçamento no tempo

O entrelaçamento no tempo é feito com o auxílio de um entrelaçador de bloco, que entrelaça as subportadoras entre os símbolos OFDM, porém limitado para atuar separadamente entre cada segmento. A profundidade de entrelaçamento é obtida variando-se o parâmetro  $I$  da equação 3.15 [8].

$$\pi(i) = I \times (5i \bmod N_c) , \quad (3.15)$$

onde:  $I$  = Ajuste do tamanho do entrelaçador,  $i$  = Entrada do entrelaçador temporal,  $\pi(i)$  = Saída do entrelaçador temporal.

A Figura 3.18 (Modificada de [4]) mostra o resultado do entrelaçamento no tempo para  $I = 8$  de um único segmento OFDM em modo 1. Nota-se que são necessários utilizar 768 símbolos OFDM para recuperar a informação entrelaçada, ou seja, aproximadamente quatro quadros OFDM.

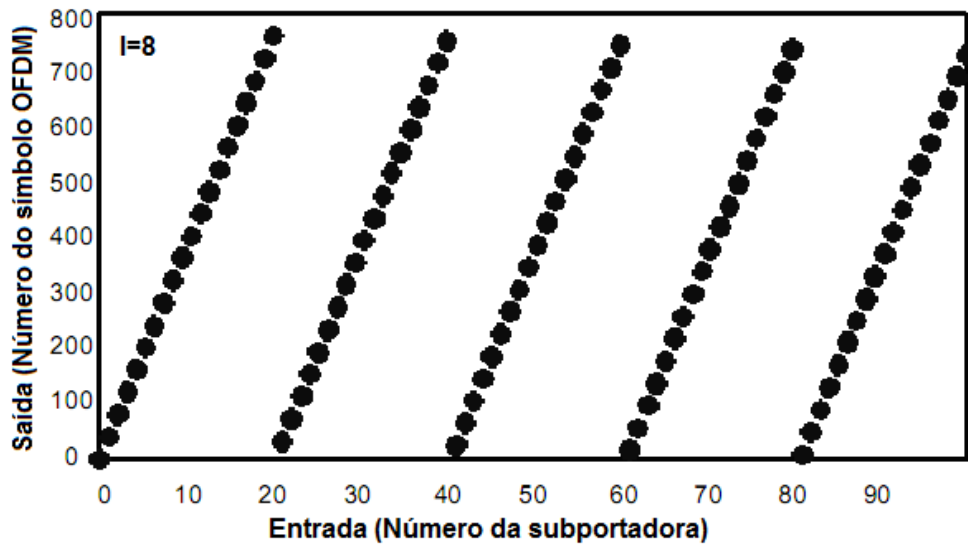


Figura 3.18: Localização das subportadoras de dados nos símbolos OFDM após o entrelaçamento no tempo para  $I = 8$  e  $Modo = 1$ .

### 3.10.4 Bloco de entrelaçamento na frequência

O último bloco do sistema é o bloco de entrelaçamento em frequência, que pode ser dividido em duas classes distintas:

- Entrelaçamento entre os segmentos;
- Entrelaçamento dentro do segmento:

Rotação de portadoras por número de segmentos;

### Aleatorização das portadoras.

O entrelaçamento entre segmentos deve ser utilizado somente quando existe mais de um segmento por camada, ou seja, os dados são entrelaçados entre os segmentos de uma mesma camada. Outro ponto importante é que o segmento destinado à recepção parcial não deve participar do entrelaçamento entre segmentos. O arranjo de símbolos antes e após este entrelaçamento, para os três modos de operação, são dados pela Figura 28 da norma [4].

No caso do entrelaçamento dentro do segmento, a rotação das portadoras é conduzida como mostra a Figura 29 de [4], enquanto a aleatorização de portadoras é dada pelas Tabelas 14, 15 e 16, respectivamente, para os modos 1, 2 e 3 também de [4].

### 3.11 Considerações finais

Neste capítulo foram vistas as etapas do sistema ISDB-T<sub>b</sub> que fazem parte da proposta deste trabalho. No próximo capítulo iremos ver os detalhes de implementação destas etapas no Matlab, onde serão feitas algumas considerações a respeito das estruturas apresentadas juntamente com adição de novas estruturas.

## CAPÍTULO 4

---

### Metodologia do simulador

---

#### 4.1 Introdução

Neste capítulo será apresentada a estrutura do simulador criado em Matlab com objetivo de gerar resultados para a criação de uma base de dados que será consultada pelo software de visualização de resultados. Os parâmetros que o simulador deve possuir são:

- Taxas dos códigos convolucionais ( $1/2$ ,  $2/3$ ,  $3/4$ ,  $5/6$  e  $7/8$ );
- Modulações digitais (QPSK, 16-QAM e 64-QAM);
- Modos de operação (1, 2 e 3);
- Intervalo de guarda ( $1/4$ ,  $1/8$ ,  $1/16$  e  $1/32$ );
- Profundidade de entrelaçamento (0, 1, 2, 4, 8, 16).
- Distribuição de segmentos entre as camadas:

Configuração 1: 13 segmentos para a camada “A”, zero para “B” e zero para “C”;

Configuração 2: 1 segmento para a camada “A”, 12 para “B” e zero para “C”;

Configuração 3: 1 segmento para a camada “A”, 3 para “B” e 9 para “C”.

Note que a modulação  $\pi/4$ -DQPSK não será implementada. Essa modulação será desprezada devido ao Matlab não possuir um método de decisão suave (*soft-decision*) para a mesma. Sem um método de decisão suave, o desempenho do algoritmo de Viterbi diminui consideravelmente, fazendo com que os resultados simulados se distanciem da realidade. Porém, o simulador será criado levando em consideração uma futura implementação deste tipo de modulação, ou seja, todas as configurações previstas para modulação diferencial, vistas no Capítulo 3, serão implementadas (configuração de segmento, entrelaçamentos, portadoras piloto, entre outros). A modulação diferencial é desejada em sistemas de recepção móvel, como esse trabalho visa a recepção fixa, a falta deste tipo de modulação não afetará os resultados desejados.

O simulador deve ser capaz de transmitir qualquer tipo de dado, como por exemplo: imagens, vídeos, sequências aleatórias de bits, dentre outros. Desta forma, alguns dados são gerados pelo simulador e outros são lidos externamente, com uma estrutura pré-definida descrita na Seção 4.3.2.

O resultados das simulações serão expostos e salvos na forma de figuras e arquivos de texto contendo vetores/matrizes com os resultados. No caso de ser utilizado como fonte de dados para transmissão um arquivo externo contendo um *stream* de bytes, o resultado da simulação será salvo na mesma estrutura lida pelo simulador.

- Figuras geradas e salvas:

Contendo a configuração de segmentos e intervalo de guarda;

Contendo a estrutura dos TSPs remultiplexados (no caso de transmissão com imagens);

Imagens recebidas para os diferentes níveis de ruído do canal (no caso de serem transmitidas imagens);

Curvas de desempenho das camadas individuais;

Curvas de desempenho de todas as camadas;

- Arquivos de textos gerados e salvos:

Arquivo de texto contendo o *log* da simulação, ou seja, os passos tomados pelo simulador;

Arquivos contendo os símbolos das constelações recebidas;

Arquivos de texto contendo os TSPs recebidos (no caso de transmissão com arquivos externos).

Os itens cuja a utilização do estimador de canal se torna indispensável, serão gerados e salvos tanto para os quadros com estimação de canal quanto para os quadros sem estimação de canal.

O item “Arquivos contendo os símbolos das constelações recebidas” terá as partes imaginárias e reais dos símbolos complexos salvos em arquivos distintos. Como esses dados complexos são utilizados para a geração de vídeos das constelações, isso facilita a leitura destes dados através do software de visualização de resultados.

## 4.2 Considerações feitas a respeito do ISDB-T

Algumas considerações a respeito do transmissor e do receptor podem ser feitas com o objetivo de tornar o simulador mais simples de ser implementado:

- Sincronização ideal entre início de símbolo OFDM [24];
- Sincronização ideal de frequências de portadoras [24];
- Sincronização de intervalo de amostragem [24];

- Dados transportados pelas TMCCs 100% livre de interferências do canal;
- Byte de sincronismo do início do TSP utilizado para identificação das camadas;
- Blocos de ajuste de atrasos não utilizados: Necessários somente se o simulador estiver operando em tempo real;
- Etapas de *upconversion*, responsável por colocar o sinal na faixa de frequência que se deseja transmitir e *downconversion*, responsável por colocar o sinal em banda base novamente após a transmissão, não foram implementadas devido a esse processo não interferir nos resultados desejados;
- Utilização de arredondamentos de informação em quadros e vetores de dados: Ocorre quando se deseja transmitir uma quantidade finita de dados que não seja suficiente para preencher todo um quadro de informação ou quando um vetor não retorna um valor inteiro ao ser fracionado.

Os blocos que são relevantes para a obtenção dos resultados desejados estão contidos na Figura 4.1. Da mesma forma, pode-se visualizar os blocos do lado do receptor, trazidos pela Figura 4.2.

Nota-se que os blocos contidos no transmissor e no receptor são praticamente os mesmos, com funções inversas. Porém no receptor temos um bloco de estimação de canal. Esse bloco tem a função de achar a resposta estimada do canal visando corrigir as distorções causadas por ele.

### 4.3 Fontes de dados para transmissão

O simulador pode transmitir dados adquiridos de duas formas:

- Gerados pelo próprio simulador (Transmissão de imagens e dados aleatórios);
- Dados lidos de arquivos externos (Transmissão de vídeos, imagens, dentre outros).



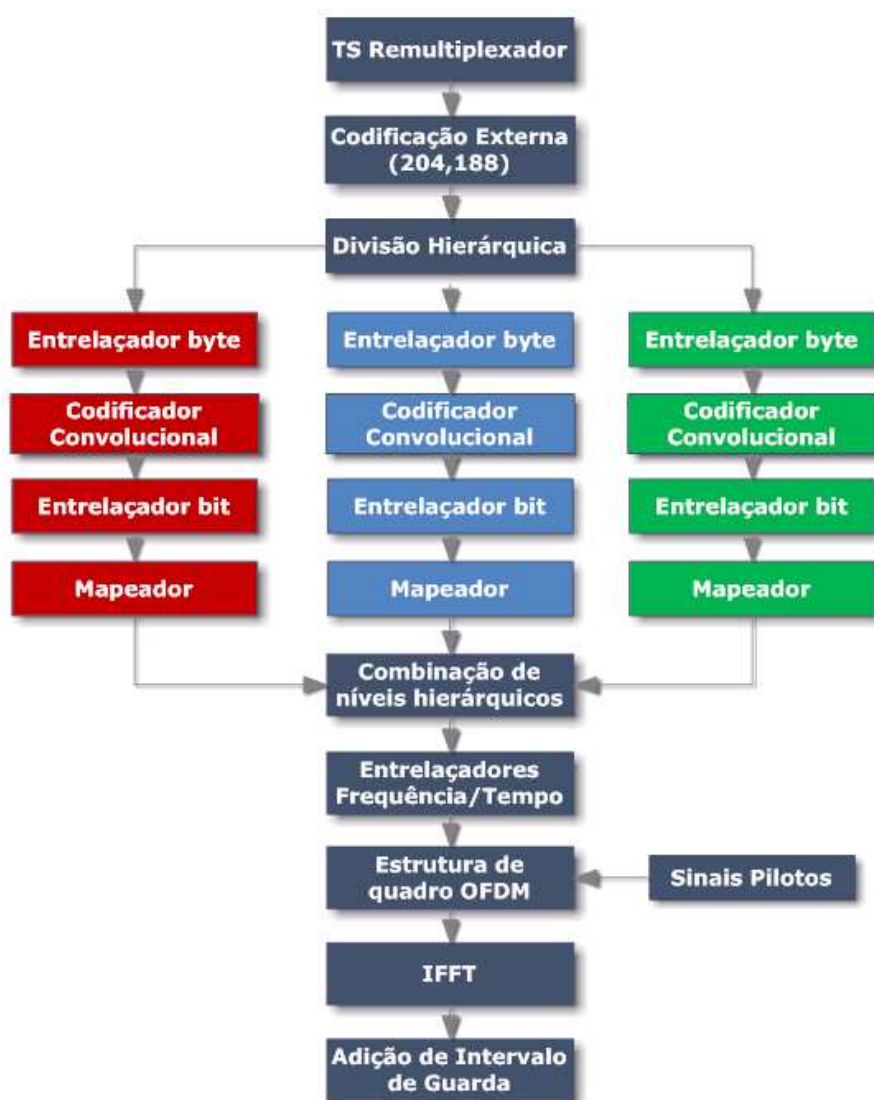


Figura 4.1: Blocos do transmissor implementados em Matlab.

### 4.3.1 Dados gerados pelo simulador

Os dados gerados pelo simulador podem ser:

- Binários aleatórios;
- Imagens TIFF e PNG.

Os dados aleatórios podem ser gerados de forma proporcional à capacidade de cada camada. Desta forma, o simulador calcula a quantidade de bits que serão gerados para uma quantidade específica de quadros ou para uma quantidade específica de bits, que

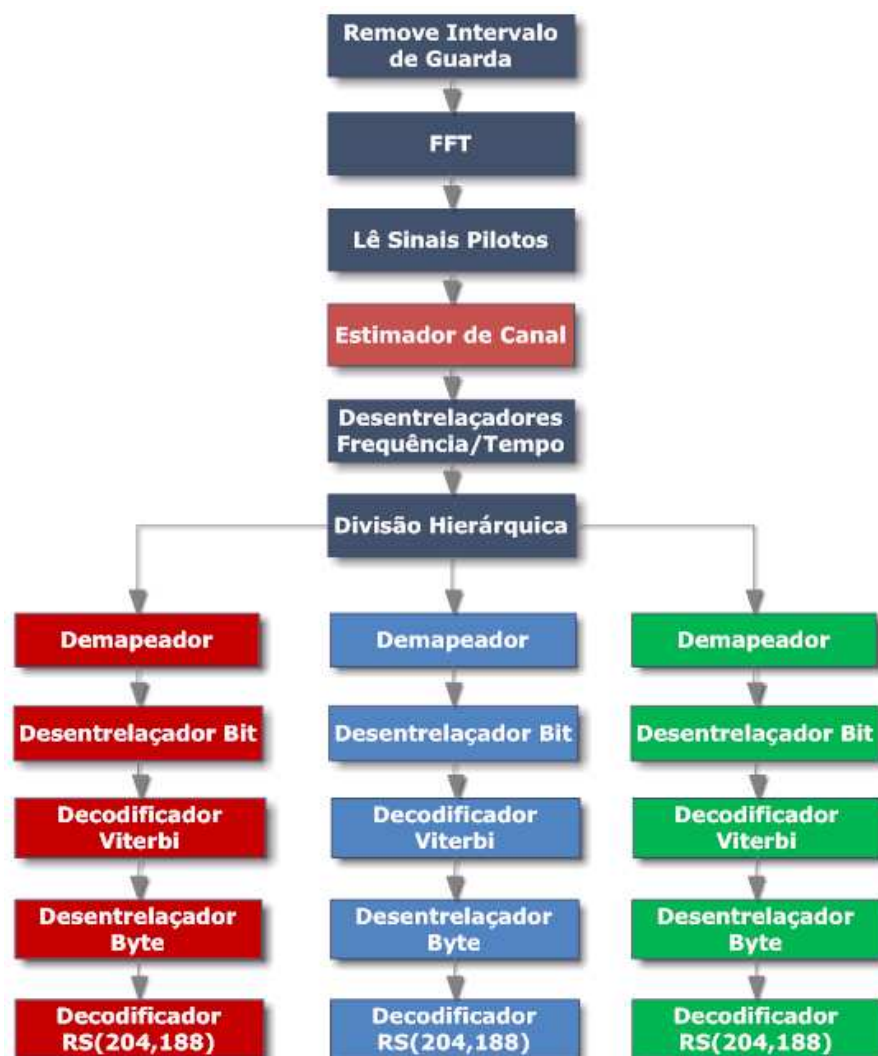


Figura 4.2: Blocos do receptor implementados em Matlab.

representa a quantidade de bits que a camada de maior capacidade de informação irá ter. Desta forma, a quantidade de dados necessários para preencher as demais camadas são calculados de maneira proporcional à capacidade das mesmas.

Para transmitir imagens TIFF e PNG, pode-se utilizar uma quantidade específica de quadros como parâmetro de quantidade de informação a ser transmitida, fazendo com que o simulador utilize apenas parte da imagem. Isso é feito visando não ultrapassar a quantidade de informação que pode ser transmitida para a quantidade de quadros especificada.

Após carregados os dados das imagens de cada camada, o simulador divide a informação em TSPs e realiza a remultiplexação. O primeiro byte do TSP leva o número

da camada à qual esse TSP pertence. Isso pode ser feito devido ao byte de sincronismo não ser utilizado pelo simulador criado.

### 4.3.2 Dados de arquivos externos

Com o intuito de tornar o simulador capaz de transmitir qualquer tipo de dados, o mesmo possui a opção de ler dados de arquivos externos, que podem ser provenientes de vídeos, imagens, sons, entre outros. Esses dados devem obedecer uma estrutura pré-definida, onde cada linha do arquivo representa um TSP, e cada coluna representa um byte do TSP. A Figura 4.3 mostra duas matrizes de dados lidas de arquivos externos. A matriz superior mostra uma estrutura de dados sem codificação externa, contendo apenas 188 colunas, enquanto a matriz inferior mostra uma estrutura já codificada. Caso os TSPs não tenham sido codificados, o próprio simulador irá codificá-los com o código RS(204, 188), se essa opção estiver habilitada.

Estrutura SEM codificação externa

1	2	3	4
1	162	124	135
1	155	127	128
1	154	134	132
1	158	134	125
1	171	131	126
1	129	131	126
1	102	134	117
1	104	135	121
1	109	132	158
1	113	132	153

...

187	188
145	118
141	109
145	123
150	96
168	82
182	85
206	124
210	58
74	56
95	0

Estrutura COM codificação externa

1	2	3	4
1	162	124	135
1	155	127	128
1	154	134	132
1	158	134	125
1	171	131	126
1	129	131	126
1	102	134	117
1	104	135	121
1	109	132	158
1	113	132	153

...

203	204
237	79
79	247
25	190
70	72
186	23
44	245
185	233
116	36
82	178
132	142

Figura 4.3: Dados importados de arquivos externos.

A Figura 4.4 apresenta a mesma estrutura de dados codificados mostrados pela Figura 4.3, porém dentro de um arquivo de extensão “.txt”.

1	162	124	135		237	79
1	155	127	128		79	247
1	154	134	132		25	190
1	158	134	125		70	72
1	171	131	126		186	23
1	129	131	126	...	44	245
1	102	134	117		185	233
1	104	135	121		116	36
1	109	132	158		82	178
1	113	132	153		132	142

Figura 4.4: Estrutura de dados de um arquivo de texto externo.

Lembre-se que a primeira coluna de dados indica a camada que os TSPs pertencem, onde nas Figuras 4.3 e 4.4 todos os TSPs pertencem à camada A, representada pelo número “1”. Note que os bytes são separados uns dos outros através de um espaço simples.

#### 4.4 Estrutura do simulador

Busca-se como resultados de uma simulação, curvas de desempenho, imagens e vídeos das constelações, para tornar possível a visualização dos impactos causados pelo canal de transmissão ao sinal recebido. Para se obter esses resultados, existem algumas estruturas que foram adicionadas ao sistema juntamente com blocos de decisões (IF) e estrutura de repetições (FOR). O funcionamento do simulador pode ser entendido analisando o fluxograma que está dividido entre as Figuras 4.5, 4.9 e 4.11.

Na primeira etapa, representada pela Figura 4.5, os parâmetros a serem utilizados na simulação já foram escolhidos: configuração, modo de transmissão, modulações, taxas de codificação, modelo de canal, entre outros. Deve-se definir se serão transmitidos dados externos ou dados gerados pelo próprio simulador, como apresentado na Seção 4.3. Seguindo o fluxograma, se a opção de codificação externa estiver habilitada, a mensagem será codificada com o código RS(204, 188).

Após a codificação externa, os TSPs sofrem a divisão hierárquica, ou seja, o simulador separa os dados em camadas de acordo com o índice 1, 2 ou 3 localizados na posição de primeiro byte do TSP. Já separados em camadas, os dados passam pelo bloco de entrelaçamento de byte e posteriormente são convertidos de decimal para binário.

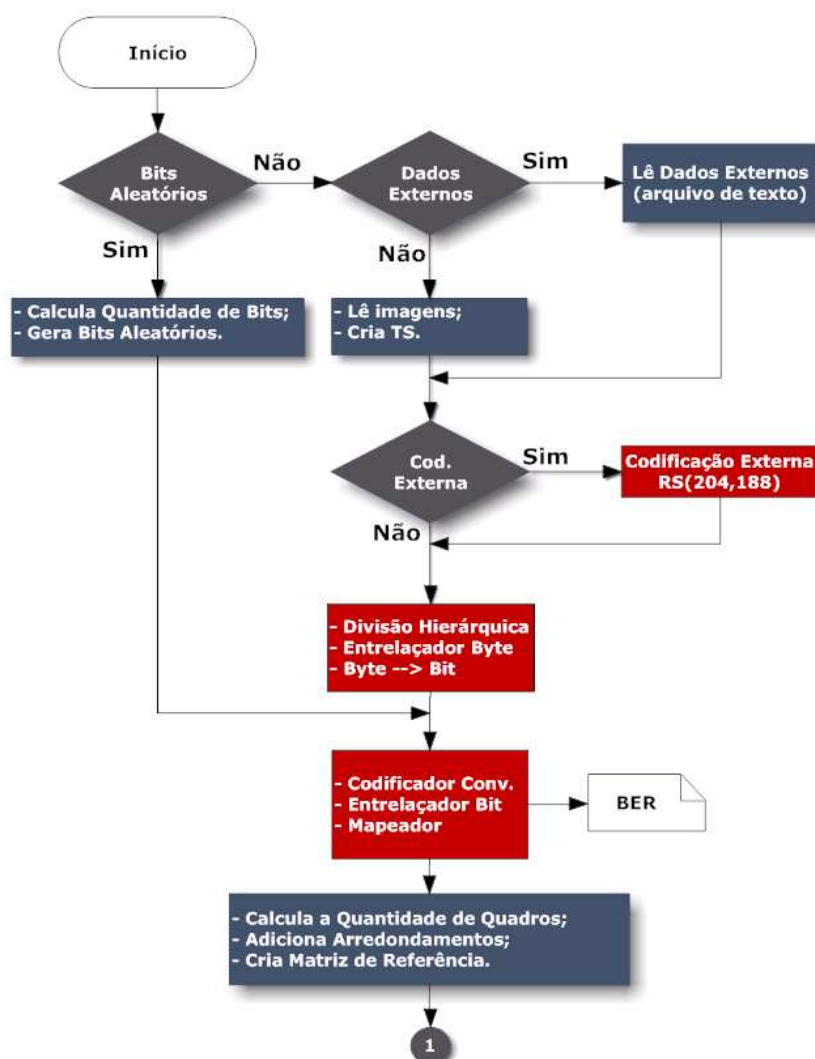


Figura 4.5: Fluxograma do simulador (parte 1).

Note que no caso de transmissão de dados aleatórios binários, os passos descritos no parágrafo anterior, bem como a aplicação da codificação externa, não são realizados. Ou seja, para a geração de dados binários aleatórios, o tratamento da informação acontece a partir deste momento, como ilustra o fluxograma. Assim, todas as etapas de tratamento da informação a nível de byte é poupada, o que torna a simulação mais rápida. Com isso, o cálculo da BER dos códigos concatenados (RS + Código Convolutacional) será feita analiticamente com base na BER calculada após a decodificação do código convolutacional. Isso se dá devido a mensagem binária aleatória não passar pelo processo de codificação externa.

O próximo passo consiste em realizar a codificação interna, que irá aumentar o

tamanho das sequências de dados de cada camada. Realiza-se também o entrelaçamento e mapeamento dos bits em símbolos para cada uma das camadas hierárquicas. Uma observação importante a ser feita é que os dados na entrada do codificador convolucional são utilizados posteriormente para o cálculo da BER.

Em seguida, com os dados já mapeados, é calculada a quantidade de quadros necessária para realizar a transmissão. Se necessário, são inseridos alguns arredondamentos aos dados de cada camada. Nesta etapa do fluxograma, é criada uma matriz que será usada como referência posteriormente. A matriz de referência tem as mesmas dimensões do quadro OFDM, e através de representações numéricas, indica as posições que irão ser ocupadas por subportadoras e por portadoras piloto. Note que o tipo de modulação (diferencial ou síncrona) e o modo de operação do sistema influenciam nas posições das portadoras nos segmentos, que por sua vez, alteram a estrutura final do quadro OFDM. Cada tipo de portadora piloto tem uma representação numérica na matriz de referência. Assim, através de um *script* em Matlab, pode-se criar uma figura que mostre as estruturas dos segmentos e dos quadros gerados pelo simulador. Como exemplo podem ser vistas as Figuras 4.6 e 4.7, que representam um segmento OFDM, em modo 1, operando com modulação diferencial e síncrona, respectivamente.

A visualização gráfica da matriz de referência de um quadro OFDM inteiro é mostrado na Figura 4.8. Nesta figura as portadoras de dados estão em branco, nos dando uma boa noção da quantidade de portadoras piloto que compõem o quadro. Esse quadro está utilizando o método de configuração 3, onde o segmento central está utilizando modulação diferencial, e os demais modulação síncrona. As portadoras de dados da segunda camada estão na cor cinza para facilitar a visualização da configuração de segmento das três camadas.

Observando o início do fluxograma da Figura 4.9, vemos que os dados que saem do bloco de arredondamento de quadros caem dentro de uma estrutura de repetição. Tal estrutura varia de um até a quantidade de quadros OFDM necessários para passar toda a informação desejada. Desta maneira, os dados são estruturados no formato de um quadro

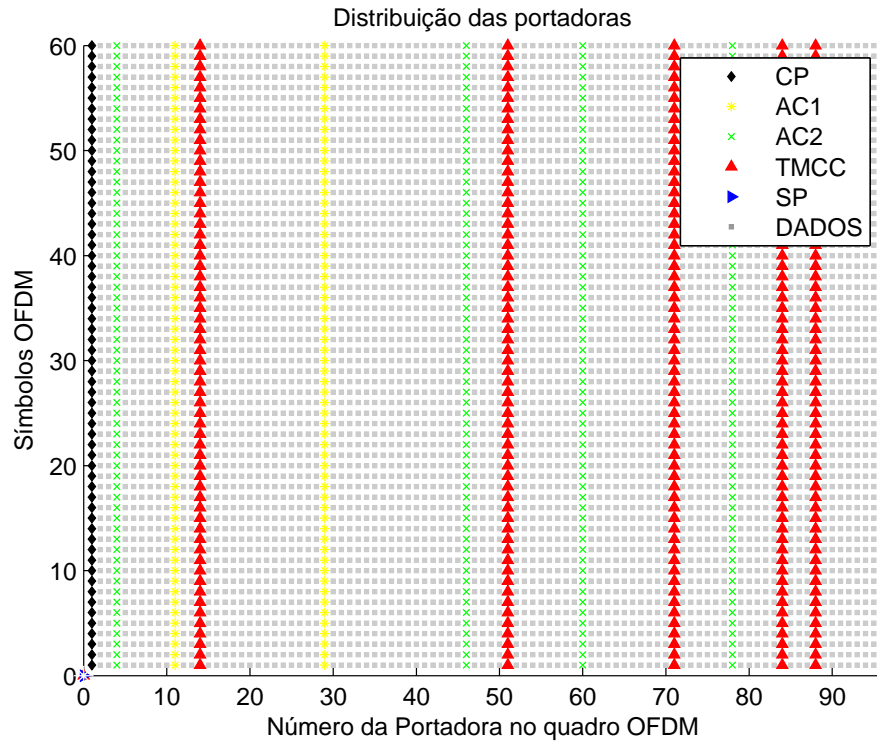


Figura 4.6: Estrutura de um segmento OFDM em modo 1, com modulação diferencial (60 símbolos OFDM e 96 subportadoras).

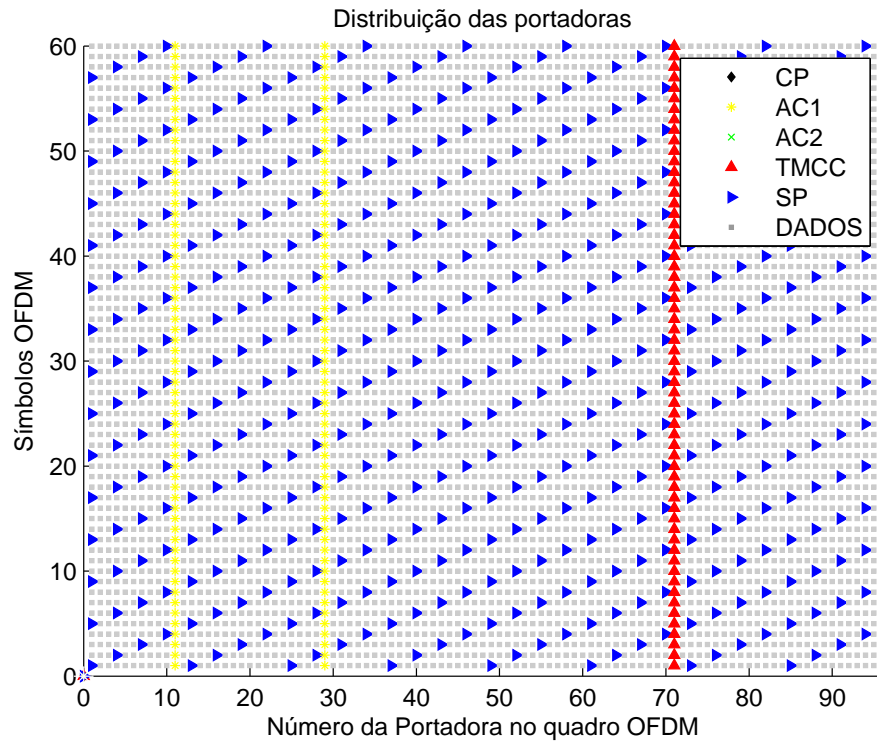


Figura 4.7: Estrutura de um segmento OFDM em modo 1, com modulação síncrona (60 símbolos OFDM e 96 subportadoras).



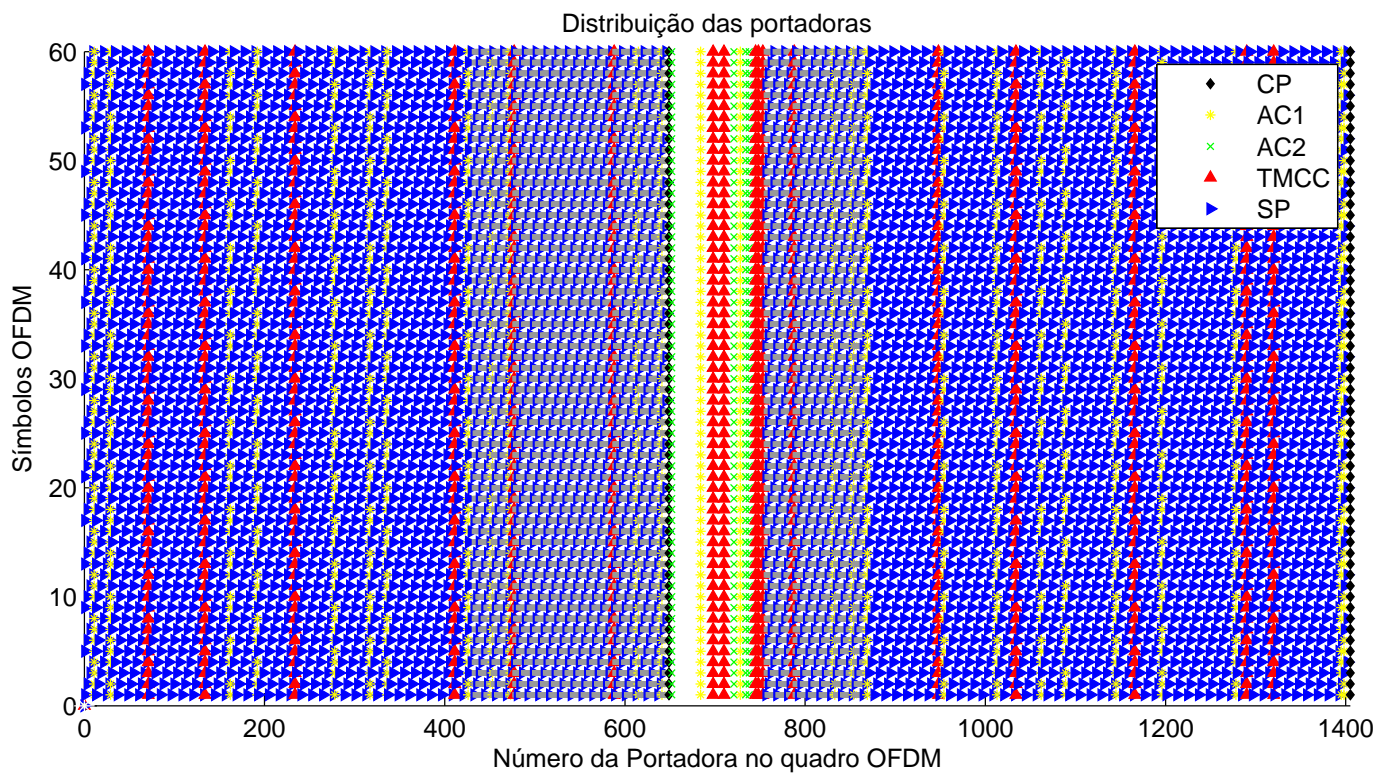


Figura 4.8: Estrutura do quadro OFDM em modo 1, configuração 3 e segmento central com modulação diferencial.



OFDM sem as portadoras piloto. Em seguida são entregues aos blocos de entrelaçamento no tempo e na frequência. Após o término do processo de entrelaçamento, são adicionadas as portadoras piloto com base na matriz de referência criada anteriormente. Os dados das portadoras piloto são gerados de forma aleatória, com exceção das TMCCs, que têm seus dados gerados conforme descrito em [4]. O conteúdo das TMCCs não é lido pelo receptor criado, uma vez que os parâmetros de transmissão são variáveis do próprio simulador, porém são gerados de acordo com a norma [4] visando futuras ampliações de projeto.

Após a inserção de portadoras piloto, é feita a IFFT de cada símbolo OFDM a uma taxa de amostragem de 2048, 4096 e 8192 amostras para os modos 1, 2 e 3, respectivamente. Note que esses valores são potências de 2, o que torna a IFFT mais eficiente do ponto de vista computacional. Um símbolo OFDM irá ter 1405, 2809 e 5617 portadoras para os modos 1, 2 e 3. Para completar o tamanho da IFFT, são adicionados zeros no centro de cada símbolo por meio de *zero-padding*.

Após a IFFT, a potência média do símbolo OFDM é normalizada. Note que as portadoras piloto tem mais energia que as portadoras de dados. Assim, a potência média das subportadoras do símbolo OFDM ( $C$ ) segue a relação [13]:

$$C = \frac{N_{dados} \times 1 + N_{pilotos} \times (4/3)^2}{N_c}, \quad (4.1)$$

onde  $N_{dados}$  representa a quantidade de portadoras de dados do símbolo OFDM e  $N_{pilotos}$  a quantidade de portadoras piloto.

Por fim, adiciona-se o prefixo cíclico a cada símbolo e em seguida a mensagem é entregue ao canal de transmissão.

O tempo de cada amostra ( $T_{ss}$ ), na saída da IFFT, após adição do prefixo cíclico, é dado por:

$$T_{ss} = \frac{T_s + T_s \times R_{IG}}{FS + FS \times R_{IG}} = \frac{T_s}{FS}, \quad (4.2)$$

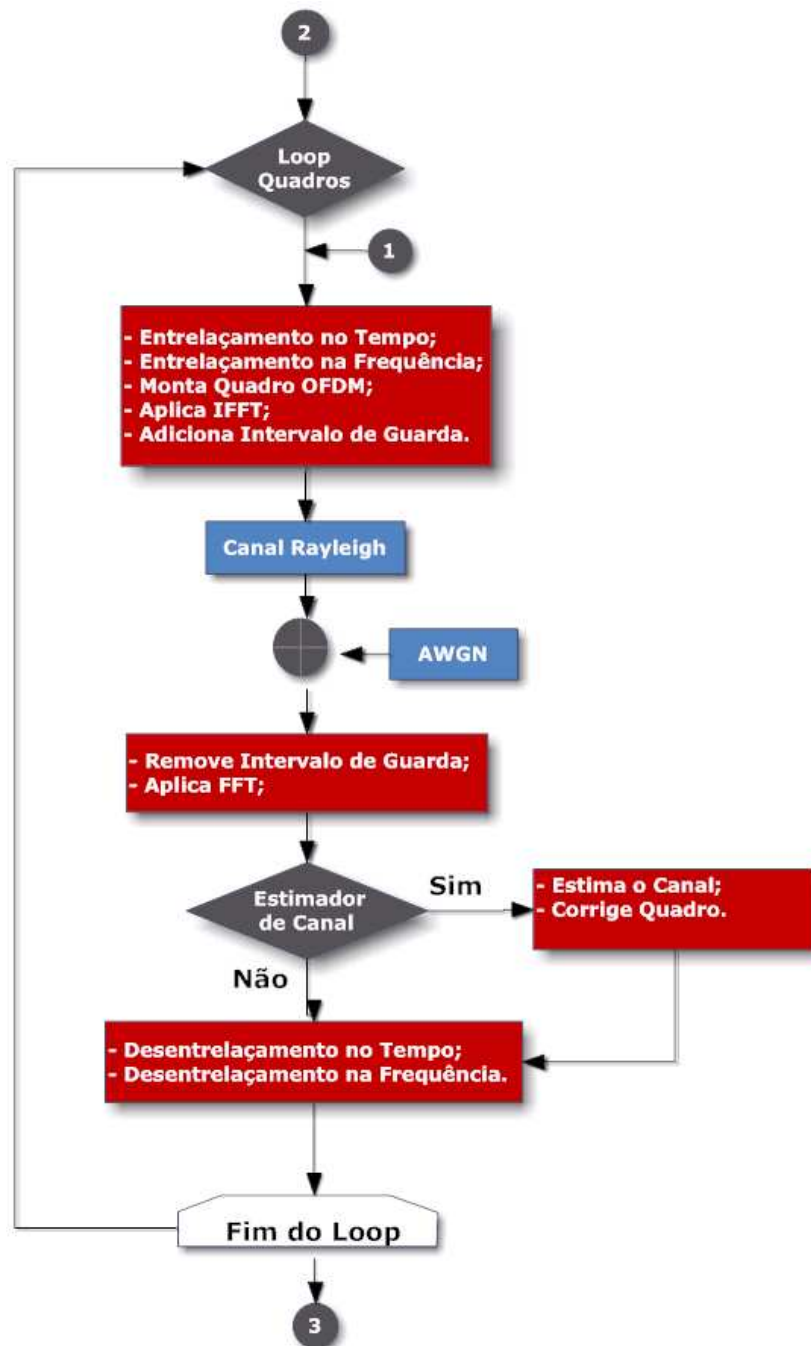


Figura 4.9: Fluxograma do simulador (parte 2).

onde  $T_s$  pode ser (252  $\mu s$ , 504  $\mu s$  ou 1008  $\mu s$ , para os modos 1,2 ou 3 respectivamente),  $FS$  é a quantidade de amostras na saída da IFFT (2048, 4096 ou 8192, para os modos 1,2 ou 3 respectivamente) e  $R_{IG}$  representa a duração do intervalo de guarda expressado como uma fração da duração do símbolo OFDM (1/4, 1/8, 1/16 ou 1/32).

Note que ao alterar o valor de  $T_s$ , a quantidade de amostras geradas na saída da

IFFT/FFT também muda.

$$T_{ss} = \frac{252\mu s}{2048} = \frac{504\mu s}{4096} = \frac{1008\mu s}{8192} = 1,2305e^{-7}, \quad (4.3)$$

Isso torna o valor de  $T_{ss}$  uma constante. Fazendo o inverso deste valor, obtém-se a frequência de amostragem ( $F_{ss}$ ) de um símbolo OFDM dada pela norma [4].

$$F_{ss} = \frac{1}{1,2305e^{-7}} = 8126984Hz. \quad (4.4)$$

Como consequência da aplicação da técnica de *zero-padding* no domínio da frequência, para realizar a IFFT com um valor que seja potência de dois (2048, 4096 ou 8192), ocorre um aumento na quantidade de amostras geradas no domínio do tempo. Essas amostras extras devem ser descartadas antes da transmissão para que a largura de banda do canal seja respeitada. Posteriormente as amostras descartadas são recuperadas por interpolação, no receptor. Nas simulações feitas em Matlab, não se observou um ganho computacional significativo que justifique a aplicação do *zero-padding*. Desta forma, o comprimento da IFFT utilizado nas simulações é igual a quantidade de subportadoras do símbolo OFDM: 1405, 2809 e 5617 para os modos 1, 2 e 3 respectivamente. Com isso, foi poupada a remoção das amostras extras geradas pela IFFT, bem como a aplicação de um método de interpolação para recuperar tais amostras do lado do receptor. Neste caso, quando o número de amostras não é uma potência de dois, o Matlab utiliza a IDFT.

Desta forma, através da Equação 4.2, pode-se calcular um novo tempo de amostragem que irá ser utilizado pelo simulador. Assim, para o modo 1, 2 e 3 obtém-se os valores  $1,7936e^{-7}$ ,  $1,7942e^{-7}$  e  $1,7946e^{-7}$  respectivamente.

Pode-se calcular a largura de banda do canal que será ocupada pelo sinal transmitido para cada modo de operação através de:

$$B_W = \frac{1}{T_{ss}}, \quad (4.5)$$

resultando em  $1/1,7936e^{-7} = 5575000$  Hz,  $1/1,7942e^{-7} = 5573000$  Hz e  $1/1,7946e^{-7} = 5572000$  Hz respectivamente para cada modo de operação.

Após a mensagem passar pelo canal com desvanecimento Rayleigh, com parâmetros que serão descritos em 4.5, é adicionado ruído branco gaussiano. A quantidade de ruído a ser adicionada ao sinal, pela função *awgn* do Matlab, é dada por:

$$SNR(dB) = \frac{E_b}{N_0} + 10 \log_{10} (k_{bits} \times R_{cc}) + 10 \log_{10} \left( \frac{N_c}{N_{amost}} \right), \quad (4.6)$$

onde  $SNR(dB)$  especifica a relação sinal-ruído, por amostras, do sinal e  $N_{amost}$  é a quantidade de amostras do símbolo OFDM. A última parcela da equação é responsável por corrigir a quantidade de erro que será introduzida no sinal devido ao aumento da quantidade de amostras, que representam um símbolo OFDM, causado pela adição do intervalo de guarda, *zero padding* (caso seja utilizado) e devido ao aumento de amostras decorrentes do intervalo de guarda.

Agora pode-se variar  $E_b/N_0$  para obter o nível de ruído que deseja-se adicionar na entrada do receptor. Porém, os valores de  $k_{bits}$  e  $R_{cc}$  serão únicos somente se o sistema estiver utilizando apenas uma camada (configuração 1). Para resolver esse problema, o simulador faz uma relação entre os valores de  $E_b/N_0$  de cada camada. Para melhor entender o que isso representa, pode-se supor que estão sendo utilizadas as três camadas do sistema, com os seguintes valores de  $k_{bits}$  para as três camadas:  $k_1 = 2$ ,  $k_2 = 4$  e  $k_3 = 6$ , e as taxas dos códigos convolucionais  $R_{cc1} = 1/2$ ,  $R_{cc2} = 2/3$  e  $R_{cc3} = 3/4$ , respectivamente, para as camadas “A”, “B” e “C”.

Desta forma,  $R_{cc1}k_1 = 1$ ,  $R_{cc2}k_2 = 2,66$  e  $R_{cc3}k_3 = 4,5$  e convertendo esses valores para dB, se obtém 0dB, 4,15dB e 6,53dB para as camadas “A”, “B” e “C”, respectivamente. O simulador irá utilizar na Equação 4.6 o menor valor encontrado, ou seja 0dB, e irá subtrair esse valor dos outros dois encontrados. Isso significa que ao entrarmos com  $E_b/N_0$  igual a 1 na Equação 4.6, será obtido um valor de SNR que representa exatamente  $E_b/N_0 = 1$  para a camada A, porém para a camada B, esse valor mensurado representa

$E_b/N_0 = 1 - 4,15 = -3,15\text{dB}$ . Para a camada C, o ruído mensurado pela função irá representar  $E_b/N_0 = 1 - 6,53 = -5,53\text{dB}$ .

Com isso, encontramos uma relação que permite avaliar o desempenho das três camadas como mostra a Figura 4.10.

A(0dB)	B(-4,15dB)	C (-6,53dB)
		-5.53
		-4.53
		-3.53
	-3.15	-2.53
	-2.15	-1.53
	-1.15	-0.53
	-0.15	0.47
1	0.85	1.47
2	1.85	2.47
3	2.85	3.47
4	3.85	4.47
5	4.85	5.47
6	5.85	6.47
7	6.85	7.47
8	7.85	8.47
9	8.85	9.47
10	9.85	10.47
11	10.85	11.47
12	11.85	12.47
13	12.85	13.47
14	13.85	
15	14.85	
16	15.85	
17		
18		
19		
20		

Figura 4.10: Exemplo de relação entre  $E_b/N_0$  para as três camadas com diferentes parâmetros.

Seguindo com a descrição do fluxograma, após a mensagem passar pelo canal e sofrer adição de ruído, ela chega ao receptor. A primeira etapa do processo é a remoção do intervalo de guarda, em seguida é feita a FFT de cada símbolo OFDM, onde as sequências de zeros adicionados pelo processo de *zero padding* são removidas (quando utilizada essa técnica).

Dando sequência ao fluxograma, se o estimador de canal não estiver habilitado a mensagem recebida é reestruturada. Nesse processo são removidas as portadoras piloto, deixando a mensagem pronta para ser desentrelaçada na frequência e no tempo. Caso o estimador de canal esteja habilitado, antes do processo de desentrelaçamento na frequência e no tempo, o quadro OFDM é reestruturado e dividido pela resposta estimada do canal.

A resposta estimada do canal é obtida utilizando um estimador de duas dimensões com método de interpolação *Cubic-Spline*. Esse processo será abordado na Seção 4.6.

Existe ainda a opção de duplicar a mensagem que chega ao bloco de decisão “Estimador de canal”, e realizar os dois processos. Dessa forma é feito o desentrelaçamento no tempo e na frequência para o quadro estimado e para o quadro não estimado. Isso serve para analisar o desempenho e a importância do bloco de estimação de canal no receptor.

Com isso chega-se ao fim da estrutura de repetição, que irá repetir o processo até que sejam passados todos os quadros OFDM. Em outras palavras, o fluxograma da Figura 4.9 será repetido até que todos os quadros necessários sejam transmitidos.

Por fim, pode-se analisar o fluxograma da Figura 4.11. Nota-se que o início e o fim do segundo fluxograma analisado se encontra interno a esse terceiro. A saída do segundo fluxograma, indicada pelo círculo de número 3, é entregue a uma outra estrutura de repetição que é responsável por repetir o processo duas vezes: uma para tratar a informação do quadro não estimado e outra para tratar o quadro estimado.

Seguindo a estrutura, são removidas as subportadoras de arredondamentos de quadros de cada uma das camadas. Então, inicia-se o processo de demapeamento em *soft-decision*, seguido pelas etapas de desentrelaçamento de bits e de decodificação das mensagens pelo algoritmo de Viterbi com método de decisão em *soft-decision*. Note que nesse ponto os dados da saída do decodificador são utilizados para calcular a BER na recepção.

Se os bits são gerados de maneira aleatória pelo simulador, o processo termina nesse ponto, desprezando os blocos que se encontram internos ao retângulo pontilhado em verde da Figura 4.11. Ao concluir a estrutura de repetição responsável por tratar os dados estimados e não estimados, todo o processo do segundo fluxograma é repetido devido a estrutura de repetição de níveis de ruídos. Assim, todos os quadros são transmitidos novamente para cada valor de ruído que se deseja simular ( $E_b/N_0$ ). Ao final de cada repetição é calculada a taxa de erro de bits. Se  $E_b/N_0$  varia de 1 até 20 dB, teremos 40 valores de BER, onde 20 serão para o quadro estimado e 20 para o tratamento da informação sem estimador.

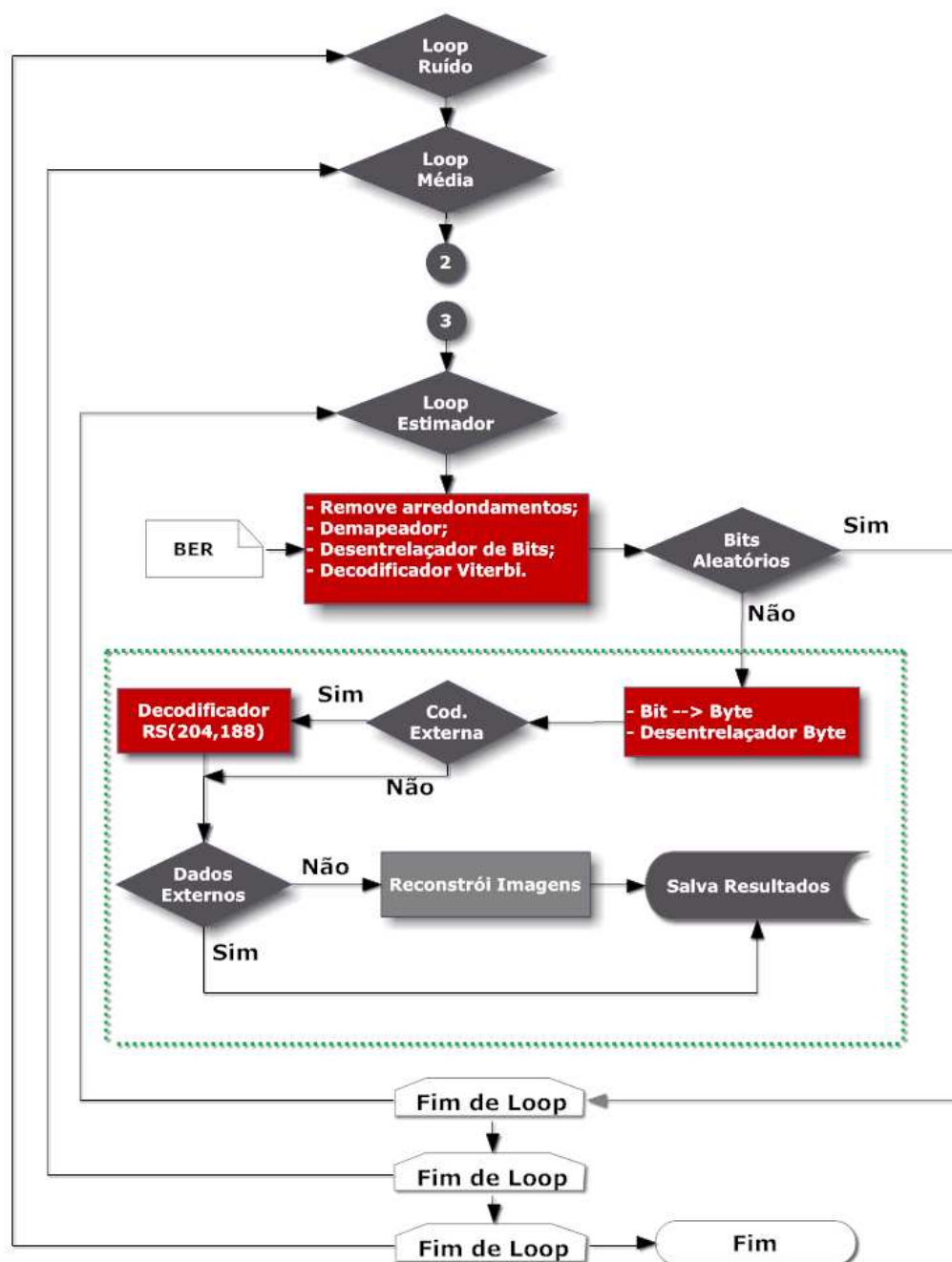


Figura 4.11: Fluxograma do simulador (parte 3).

Analisando o terceiro fluxograma, pode-se notar que entre a estrutura de repetição de nível de ruído e o círculo de número dois (que marca o início do segundo fluxograma), existe uma outra estrutura de repetição. Esta estrutura, “Loop Média”, permite que a BER de cada nível de ruído possa ser calculada mais de uma vez, resultando em um BER médio. Isso é necessário para que a curva de desempenho não varie para cada simulação. Esse procedimento é desnecessário no caso de o canal de transmissão ser Gaussiano, porém,

quando se utiliza o canal de desvanecimento Rayleigh, é necessário repetir a simulação para cada nível de ruído mais de uma vez. Esse procedimento simula a transmissão do quadro para várias condições do modelo de canal utilizado, pois para cada repetição são calculados novos ganhos para os caminhos de multipercursos do modelo em uso. A troca dos ganhos dos caminhos (multipercursos) também é feita de quadro para quadro. Porém, em simulações em que a quantidade de quadros OFDM é muito pequena, deve-se utilizar um valor suficiente de repetições para estabilizar a curva de desempenho.

A taxa de erro de bits após a codificação externa é estimada analiticamente, tendo como base de cálculo a BER obtida após a decodificação interna. Esse procedimento torna o simulador mais simples de ser implementado, poupando outros pontos de cálculo de taxas de erro de bit no sistema. Outra vantagem obtida é a possibilidade de que sejam traçadas curvas de desempenho do sistema como um todo, mesmo trabalhando com a geração de dados binários aleatórios, onde o processo de codificação externa não é realizado pelo simulador. A BER após a codificação externa é obtida por:

$$BER = \sum_{e=t+1}^n \binom{n}{e} (BER_{cc})^e (1 - BER_{cc})^{n-e} , \quad (4.7)$$

onde  $e$  representa a quantidade de símbolos da palavra código que foram corrompidos por ruídos. Essa relação pode ser feita devido ao processo de entrelaçamento que garante a aleatorização dos erros.

Se a mensagem não for do tipo binária aleatória, ainda existem alguns passos antes do final das estruturas de repetições de quadro não estimado/estimado e de níveis de ruído. As mensagens nas saídas dos decodificadores de Viterbi de cada camada são entregues ao bloco de conversão de bit para byte e passam pelo bloco de desentrelaçamento de byte. Se a codificação externa estiver habilitada, as mensagens seguem para o decodificador RS(204,188) e chegam a mais um bloco de decisão. Se a mensagem transmitida não foi gerada pelo simulador, ou seja, se foi obtida através de um arquivo externo, os dados recebidos são salvos na mesma estrutura em que foram lidos. Caso a mensagem tenha sido gerada através de imagens, as mesmas são reconstruídas e salvas.



Cada valor de relação sinal-ruído, pelo qual a mensagem é submetida, gera subpastas que irão conter os resultados obtidos. Essas pastas recebem como nome o número da repetição em que a estrutura de repetição de nível de ruído se encontra. Estas pastas ficam internas a uma pasta criada no início da simulação, fazendo com que cada simulação realizada tenha uma pasta própria.

Como cada valor de SNR representa valores distintos de  $E_b/N_0$  para cada camada, o simulador cria um arquivo texto contendo informações de nível de ruído para cada camada hierárquica. Cada linha deste arquivo, chamado de “folders.txt”, representa uma repetição realizada pelo “FOR” de nível de ruído, como mostra a Figura 4.12.

folders.txt - Bloco de notas

Arquivo Editar Formatar Exibir Ajuda

Camada A: Eb/No= 1 dB;	Camada B: Eb/No= -2 dB;	Camada C: Eb/No= -4.2 dB
Camada A: Eb/No= 2 dB;	Camada B: Eb/No= -1 dB;	Camada C: Eb/No= -3.2 dB
Camada A: Eb/No= 3 dB;	Camada B: Eb/No= 0 dB;	Camada C: Eb/No= -2.2 dB
Camada A: Eb/No= 4 dB;	Camada B: Eb/No= 1 dB;	Camada C: Eb/No= -1.2 dB
Camada A: Eb/No= 5 dB;	Camada B: Eb/No= 2 dB;	Camada C: Eb/No= -0.2 dB
Camada A: Eb/No= 6 dB;	Camada B: Eb/No= 3 dB;	Camada C: Eb/No= 0.8 dB
Camada A: Eb/No= 7 dB;	Camada B: Eb/No= 4 dB;	Camada C: Eb/No= 1.8 dB
Camada A: Eb/No= 8 dB;	Camada B: Eb/No= 5 dB;	Camada C: Eb/No= 2.8 dB
Camada A: Eb/No= 9 dB;	Camada B: Eb/No= 6 dB;	Camada C: Eb/No= 3.8 dB
Camada A: Eb/No= 10 dB;	Camada B: Eb/No= 7 dB;	Camada C: Eb/No= 4.8 dB
Camada A: Eb/No= 11 dB;	Camada B: Eb/No= 8 dB;	Camada C: Eb/No= 5.8 dB
Camada A: Eb/No= 12 dB;	Camada B: Eb/No= 9 dB;	Camada C: Eb/No= 6.8 dB
Camada A: Eb/No= 13 dB;	Camada B: Eb/No= 10 dB;	Camada C: Eb/No= 7.8 dB
Camada A: Eb/No= 14 dB;	Camada B: Eb/No= 11 dB;	Camada C: Eb/No= 8.8 dB
Camada A: Eb/No= 15 dB;	Camada B: Eb/No= 12 dB;	Camada C: Eb/No= 9.8 dB
Camada A: Eb/No= 16 dB;	Camada B: Eb/No= 13 dB;	Camada C: Eb/No= 10.8 dB
Camada A: Eb/No= 17 dB;	Camada B: Eb/No= 14 dB;	Camada C: Eb/No= 11.8 dB
Camada A: Eb/No= 18 dB;	Camada B: Eb/No= 15 dB;	Camada C: Eb/No= 12.8 dB
Camada A: Eb/No= 19 dB;	Camada B: Eb/No= 16 dB;	Camada C: Eb/No= 13.8 dB
Camada A: Eb/No= 20 dB;	Camada B: Eb/No= 17 dB;	Camada C: Eb/No= 14.8 dB
Camada A: Eb/No= 21 dB;	Camada B: Eb/No= 18 dB;	Camada C: Eb/No= 15.8 dB
Camada A: Eb/No= 22 dB;	Camada B: Eb/No= 19 dB;	Camada C: Eb/No= 16.8 dB
Camada A: Eb/No= 23 dB;	Camada B: Eb/No= 20 dB;	Camada C: Eb/No= 17.8 dB
Camada A: Eb/No= 24 dB;	Camada B: Eb/No= 21 dB;	Camada C: Eb/No= 18.8 dB
Camada A: Eb/No= 25 dB;	Camada B: Eb/No= 22 dB;	Camada C: Eb/No= 19.8 dB
Camada A: Eb/No= 26 dB;	Camada B: Eb/No= 23 dB;	Camada C: Eb/No= 20.8 dB

Figura 4.12: Informações de nível de ruído para cada camada hierárquica.

Para facilitar a localização e indexação das pastas de simulações através do software de consulta, cada simulação tem um arquivo contendo um código que representa os parâmetros da mesma. Como exemplo de código de identificação temos a sequência “1 1 1/8 16-QAM 3/4 0 16-QAM 3/4 0 64-QAM 5/6 0 1”. Esse código é escrito em um arquivo de texto criado pelo simulador, chamado de “config.txt”, onde cada sequência de caracteres espaçados representa um parâmetro, como mostra a Figura 4.13.

Existem alguns *scripts* auxiliares que não foram incluídos nos fluxogramas visualizados para não torná-los muito difíceis de serem analisados. Esses *scripts* auxiliares atuam em meio aos blocos mostrados pelas Figuras 4.5, 4.9 e 4.11, e são responsáveis por gerar e

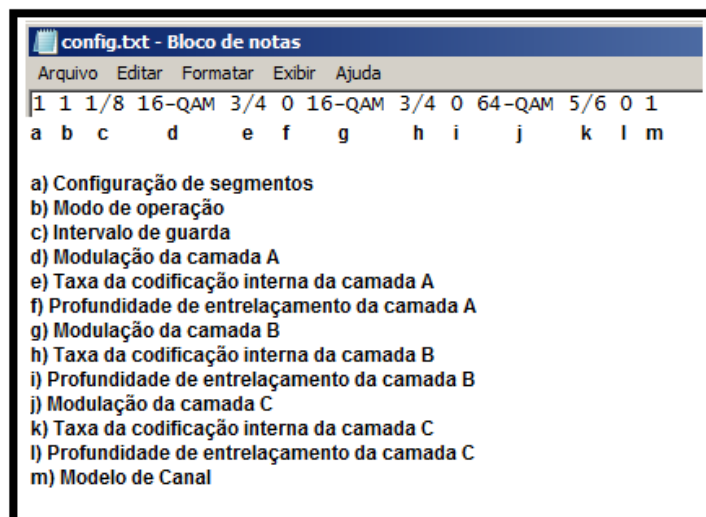


Figura 4.13: Representação do código de simulação.

salvar imagens das constelações, gerar imagens de segmentos e estrutura de quadros, criar e salvar os passos do simulador (arquivos de *log*), geração de vídeo das constelações por parte do simulador, entre outros.

Como a estrutura do simulador obedece o método de Monte Carlo, para gerar as curvas de desempenho, o simulador foi parametrizado para transmitir uma taxa de bits que garanta no mínimo 100 bits de erro, como critério de parada. Por exemplo: em simulações com modulação 64-QAM, taxa de codificação interna de 3/4 e profundidade de entrelaçamento igual a zero, são necessários gerar nove milhões de bits aleatórios para que a curva de desempenho fique estável (canal gaussiano). Note que a quantidade de bits pode ser alterada para um valor maior, porém, esse valor deve ser múltiplo da quantidade de bits que um quadro OFDM pode transmitir (que depende dos parâmetros de entrada). Isso acontece devido o simulador passar quadros inteiros pelo canal (1, 2, 3,...).

## 4.5 Modelos de canais utilizados

O simulador utiliza os modelos de canais Brasil A, B, C, D e E. Esses modelos são encontrados em diversos trabalhos, como por exemplo [25], [6], [10] e [26]. Tais modelos descrevem alguns cenários de recepção fixa, porém, podemos encontrar alguns trabalhos

em que esses modelos são utilizados juntamente com efeito Doppler para representar canais de recepção móvel.

O modelo Brasil A representa um ambiente de recepção externa que, de maneira geral, simula um sinal com múltiplos percursos de baixa intensidade e com valores de atrasos pequenos. Diferente do modelo Brasil A, o modelo Brasil B busca caracterizar um ambiente de recepção com antena interna, onde para isso são parametrizados atrasos grandes e atenuações de sinal relativamente alta, podendo até representar uma antena externa com defeitos, atingindo atenuações de até 4 dB.

O perfil Brasil C considera atrasos pequenos e de grande potência. Esse modelo pode simular a propagação em ambientes montanhosos sem linha de visada. Bem semelhante ao modelo Brasil C, o modelo Brasil D caracteriza uma recepção em ambiente com antena interna debilitada.

Por fim, o modelo Brasil E avalia um cenário de recepção com rede de frequência única. Nesse caso, o receptor estará recebendo sinais de mesma amplitude defasados no tempo entre si, de modo a simular uma recepção em um ponto onde a antena receptora encontra-se entre três transmissores em *SFN*.

A Tabela 4.1 mostra os valores de atraso e atenuações utilizados para parametrizar a função Rayleigh de cada um dos cinco modelos apresentados.

Nome	1	2	3	4	5	6
Brasil A Atraso (us)	0	0.15	2.22	3.05	5.86	5.93
Atenuação (dB)	0	13.8	16.2	14.9	13.6	16.4
Brasil B Atraso (us)	0	0.3	3.5	4.4	9.5	12.7
Atenuação (dB)	0	12	4	7	15	22
Brasil C Atraso (us)	0	0.09	0.42	1.5	2.32	2.79
Atenuação (dB)	2.8	0	3.8	0.1	2.5	1.3
Brasil D Atraso (us)	0.1	0.63	2.22	3.05	5.86	5.93
Atenuação (dB)	0.1	3.8	2.6	1.3	0	2.8
Brasil E Atraso (us)	0	1	2	-	-	-
Atenuação (dB)	0	0	0	-	-	-

Tabela 4.1: Especificação de atraso e atenuação dos modelos de canais Brasil.

A função do Matlab que utiliza a distribuição Rayleigh como modelo de canal é a

*rayleighchan*. O parâmetro referente ao tempo de amostragem do sinal que irá passar pelo canal, em segundos, para o modo 1, 2 e 3 são os mesmos calculados pela Equação 4.5.

## 4.6 Estimação de canal

É possível utilizar dois tipos de estimadores de canal para o sistema ISDB-T:

- Estimação em uma dimensão: 1D (frequência);
- Estimação em duas dimensões: 2D (frequência e tempo).

A diferença entre essas duas formas de estimação está na leitura das portadoras de referência. Essa diferença pode ser vista analisando a Figura 4.14.

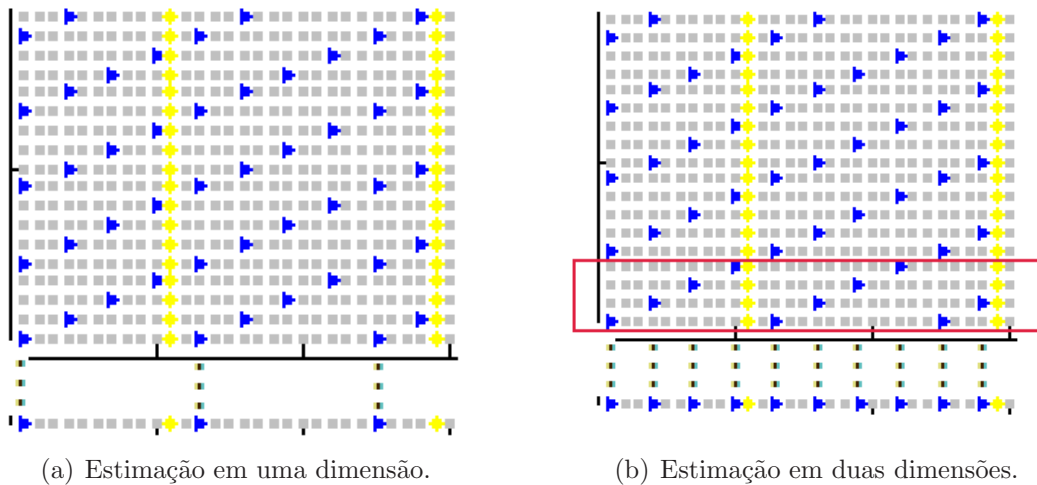


Figura 4.14: Quantidade de portadoras SPs utilizadas para estimação de canal. (a) Estimações em uma dimensão e (b) Estimações em duas dimensões.

A Figura 4.14 mostra a aquisição de portadoras SPs que serão utilizadas como referência devido ao receptor conhecer os valores destas portadoras geradas pelo transmissor, ou seja, sem a interferência do canal. Para o método de estimação 1D, são utilizadas apenas as portadoras de referência de um símbolo, Figura 4.14(a). Após a leitura destas portadoras de referências, os valores lidos são comparados com os valores conhecidos pelo

receptor, calculando a resposta estimada do canal para cada SP. A relação entre sinal transmitido (antes da IFFT no transmissor) e o sinal recebido (após a FFT no receptor) para cada subportadora é dada por:

$$Y(n_s, n_p) = X(n_s, n_p)H(n_s, n_p) + N(n_s, n_p) , \quad (4.8)$$

onde  $Y(n_s, n_p)$  e  $X(n_s, n_p)$  são respectivamente o sinal recebido e transmitido para cada subportadora do quadro OFDM.  $H(n_s, n_p)$  é a resposta do canal naquele ponto e  $N(n_s, n_p)$  é o ruído AWGN adicionado à subportadora. Os índices  $n_s$  e  $n_p$  representam respectivamente o número do símbolo OFDM no quadro e o número da subportadora (no sentido da frequência) no símbolo OFDM.

A resposta estimada do canal para cada subportadora ( $\hat{H}(n_s, n_p)$ ), pode ser calculada através de:

$$\hat{H}(n_s, n_p) = \frac{Y(n_s, n_p)}{X(n_s, n_p)} . \quad (4.9)$$

Por fim, basta realizar uma interpolação entre os pontos calculados, buscando obter uma resposta estimada do canal para as demais subportadoras, que estão localizadas entre as portadoras de referência.

Na Figura 4.14(b), são utilizadas as portadoras de referências de quatro símbolos OFDM. As portadoras de três símbolos OFDM, localizados em posições superiores no quadro OFDM, são espelhadas no símbolo em análise. Como especificado por [4], as portadoras SPs são espalhadas de 12 subportadoras no sentido da frequência e 4 portadoras no sentido dos símbolos OFDM, o que resulta em um espaçamento significativamente reduzido entre as SPs após realizar o “espelhamento” destas subportadoras sobre o eixo da frequência do símbolo que se deseja estimar, como mostrado pela Figura 4.14. Essa diminuição de espaçamento entre duas portadoras piloto torna a interpolação mais precisa, acarretando em um ganho no desempenho do estimador.

O método de interpolação utilizado é o *Cubic-Spline*, que utiliza polinômios de terceira ordem para realizar a interpolação. As interpolações por *splines* são casos especiais da interpolação *piecewise* polinomial. Esse método possui boa resposta até mesmo utilizando polinômios de baixo grau, devido à interpolação *spline* evitar o fenômeno de instabilidade numérica, que aparece quando se utilizam polinômios de grau elevado. Esse método de interpolação apresentou resultados superiores frente aos demais métodos: interpolação exponencial, *piecewise* e linear, avaliados pelos trabalhos citados no Capítulo 1 e também em [26], o que justifica a escolha deste método.

## 4.7 Considerações finais

De maneira geral, foram vistos os caminhos tomados para atingir o objetivo proposto por este trabalho. No próximo capítulo serão mostrados alguns resultados obtidos através do simulador descrito neste capítulo.

## CAPÍTULO 5

---

### Resultados

---

#### 5.1 Introdução

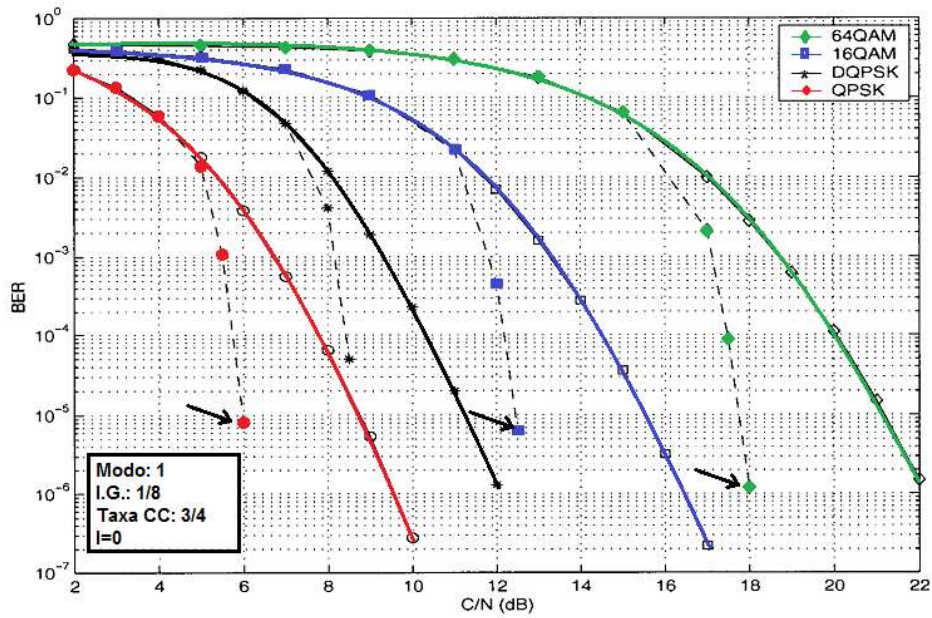
Neste capítulo serão mostrados alguns resultados gerados pelo simulador desenvolvido. Primeiramente o simulador será validado comparando resultados obtidos pelo simulador com o trabalho de [13]. Posteriormente, os resultados do método de estimação de canal adotado serão mostrados.

Dentre os resultados gerados pelo simulador, pode-se dizer que os principais são as curvas de desempenho. Estas curvas serão vistas na Seção 5.4, enquanto os demais resultados gerados pelo simulador serão abordados separadamente na Seção 5.5.

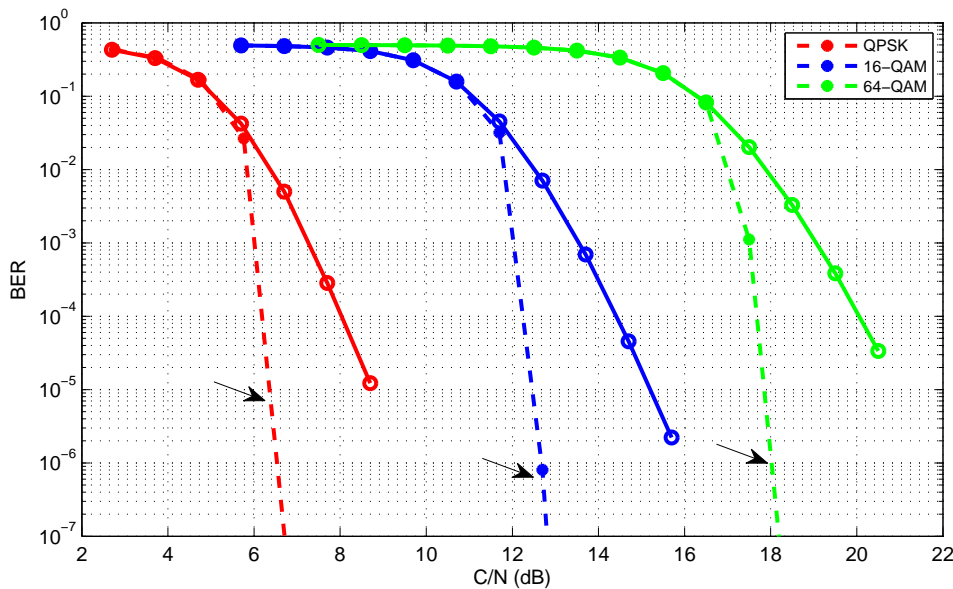
#### 5.2 Validação

Visando validar o simulador criado, pode-se comparar os resultados de algumas simulações com os resultados trazidos por [13]. Iremos comparar as curvas de desempenho das dife-

rentes modulações para os seguintes parâmetros: Taxa do  $CC = 3/4$ ;  $I = 0$ ; Intervalo de guarda igual a  $1/8$ ; Modo de operação igual a 1. A Figura 5.1(a) mostra os resultados obtidos por [13], enquanto a Figura 5.2 traz os resultados obtidos com o simulador criado.



(a) Curvas obtidas de [13].



(b) Curvas obtidas com o simulador criado.

Figura 5.1: Curvas de desempenho para as diferentes modulações com  $CC$  de  $taxa = 3/4$ , e  $CC$  de  $taxa = 3/4$  concatenado com  $RS(204,188)$ , para canais AWGN. (a) Curvas para comparação e (b) Curvas obtidas com o simulador criado.

As curvas geradas pelo simulador são  $BER \times E_b/N_0$ . Desta forma, a Figura foi alterada



para BER x  $C/N$  para facilitar a comparação entre as duas figuras. Fazendo uma relação entre  $C/N$  e  $E_b/N_0$  têm-se que as curvas geradas pelo simulador deste trabalho estarão deslocadas para esquerda a uma razão de:  $10 \log_{10}(6(3/4)) \simeq 6,5$  dB para a curva 64-QAM,  $10 \log_{10}(4(3/4)) \simeq 4,7$  dB para 16-QAM e  $10 \log_{10}(2(3/4)) \simeq 1,76$  dB para QPSK, como mostra a Figura 5.2.

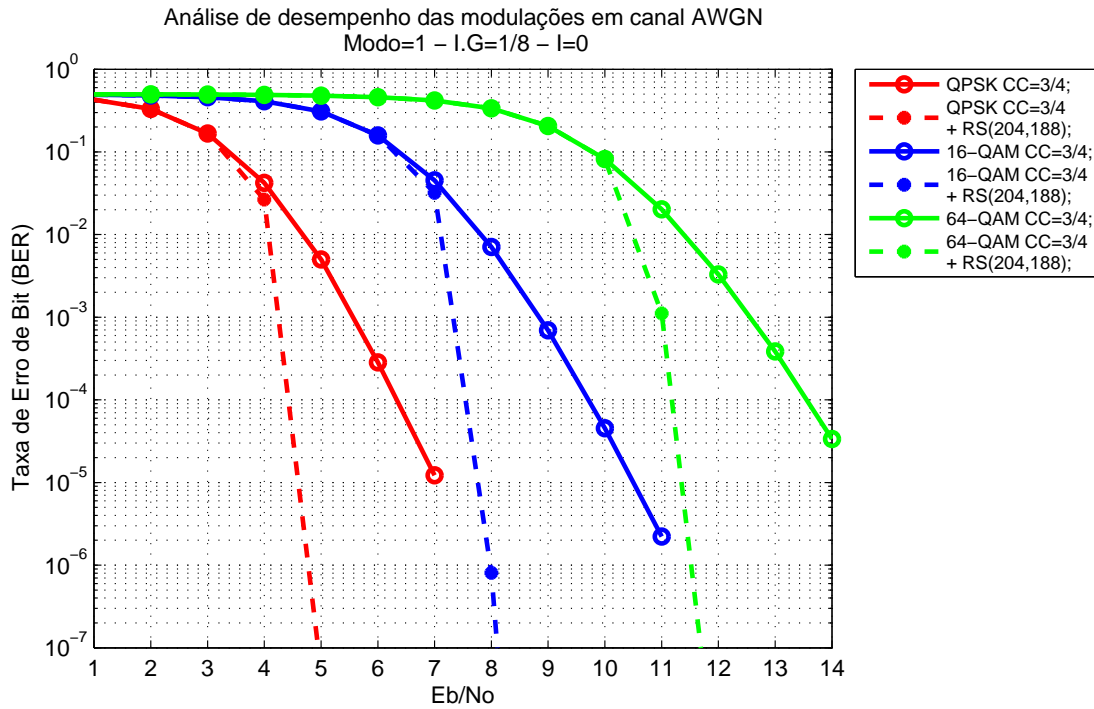


Figura 5.2: Curva de desempenho para as diferentes modulações com CC de  $taxa = 3/4$ , e CC de  $taxa = 3/4$  concatenado com RS(204,188), para canais AWGN.

Analisando a Figura 5.3, pode-se validar os resultados do simulador para o modelo de canal Brasil A. A Figura 5.3(a), retirada de [27], traz a curva do sistema ISDB-T obtida através de medições. A Figura 5.3(b) traz a curva gerada pelo simulador. Pode-se observar que o resultado simulado está bem próximo do resultado obtido por medições, o que valida a implementação do estimador e do modelo de canal.

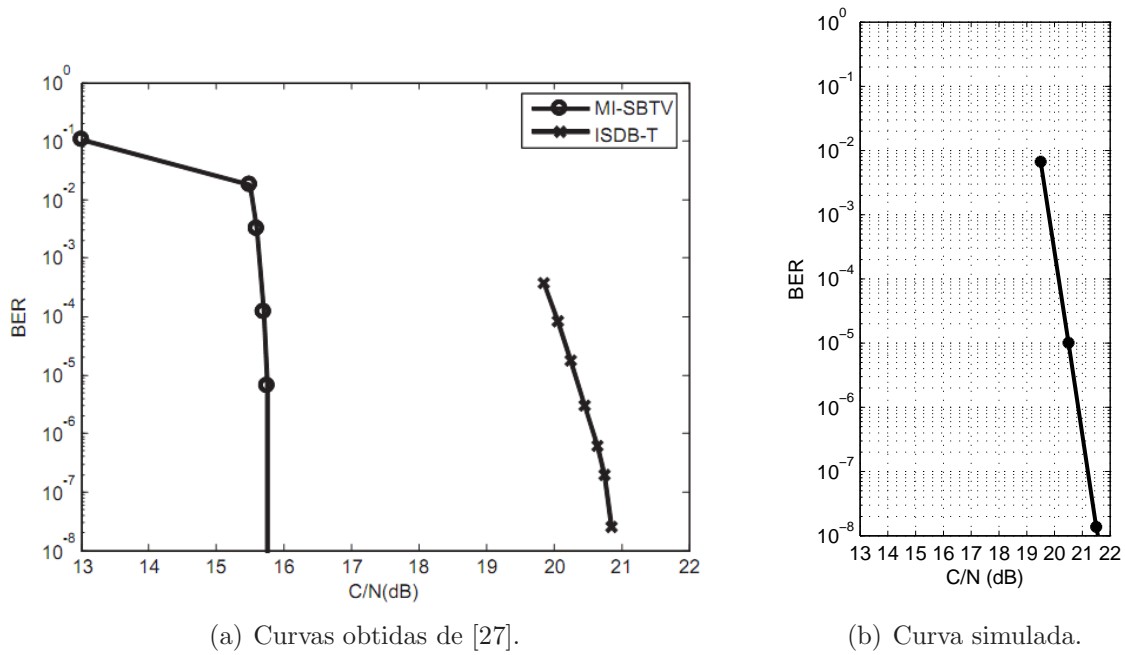


Figura 5.3: Curva de desempenho para modulação 64-QAM com CC de  $\text{taxa} = 3/4$  concatenado com RS(204,188), para canal Brasil A. (a) Curva para comparação e (b) Curva obtida com o simulador criado.

Para trazer a curva de desempenho para um valor próximo do medido, foi utilizada uma condição para os ganhos do caminho gerados pela função `rayleighchan` do Matlab. Cada vez que são gerados novos *paths gains*, é verificado se tais valores gerados geram como resultado, na saída do canal, um vetor cuja a potência média do símbolo OFDM seja aproximadamente 1,4. Esse procedimento torna as curvas geradas mais próximas dos valores medidos. Cada quadro OFDM é passado 110 vezes pelo canal, e ao final, é calculada uma média entre as taxas de erro de bit obtidas (“Loop Média” igual a 110).

### 5.3 Bloco de estimação de canal

Para analisar o desempenho do bloco de estimação, foi criado um cenário para transmissão de 1 quadro OFDM que possui suas portadoras de dados mapeadas com modulação síncrona 16-QAM, modo de operação 1, configuração 1 e taxa de intervalo de guarda igual a  $1/8$ . O modelo de canal utilizado foi o Brasil B (grandes atrasos e atenuações de sinais relativamente alta) e não foi inserido ruído AWGN (o termo  $N(n_s, n_p)$  da Equação 4.8 é

zero). Assim, os danos causados à constelação recebida são atribuídos às características de multipercurso do canal. Isso torna possível avaliar o impacto direto do estimador, pois a inserção de ruído irá prejudicar o desempenho deste. A constelação recebida sem equalização é ilustrada pela Figura 5.4.

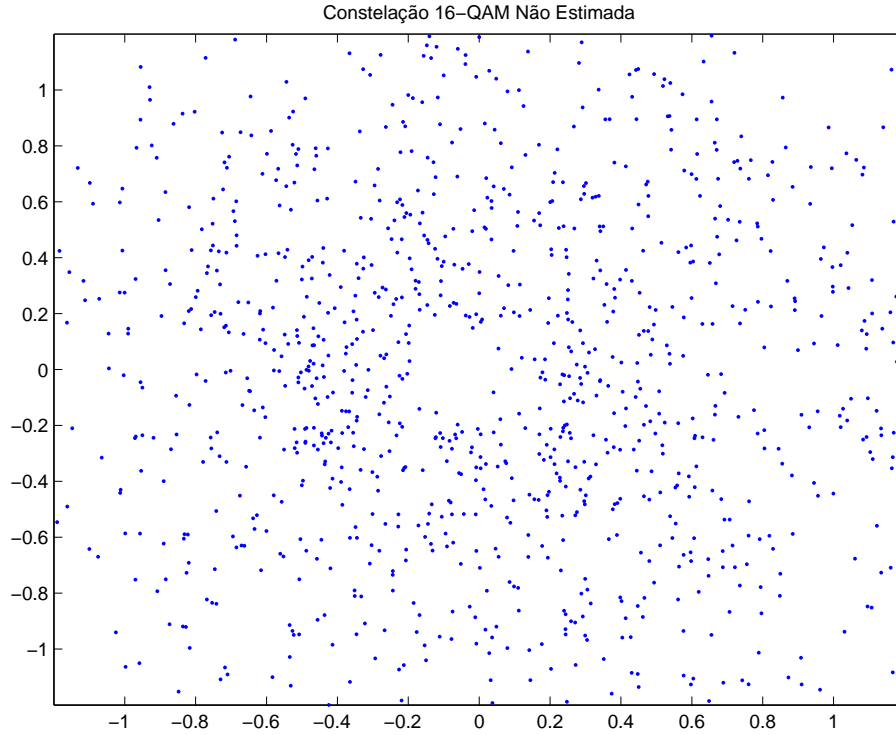


Figura 5.4: Constelação 16-QAM recebida de 1 símbolo OFDM sem estimação, corrompida por um canal de desvanecimento Rayleigh (Brasil B).

A Figura 5.4 mostra os símbolos da constelação 16-QAM recebida contidos em um símbolo OFDM inteiro, ou seja, estão plotadas todas as subportadoras de dados do mesmo.

Uma vez recebido o quadro, pode-se optar por utilizar a estimação 1D ou 2D, descritas no Capítulo 4. A resposta do canal para 1 símbolo OFDM, obtida através da estimação em uma dimensão, é dada pela Figura 5.5. Note que é feita uma comparação da resposta estimada do canal com a resposta real. A resposta real é calculada utilizando a Equação 4.9 para todas as subportadoras do símbolo OFDM.

Pode-se obter uma visão mais ampliada do resultado da interpolação através da Figura 5.6, que mostra apenas as 50 primeiras subportadoras do símbolo OFDM.

Nota-se que a distância entre as subportadoras SPs não é suficiente para garantir que

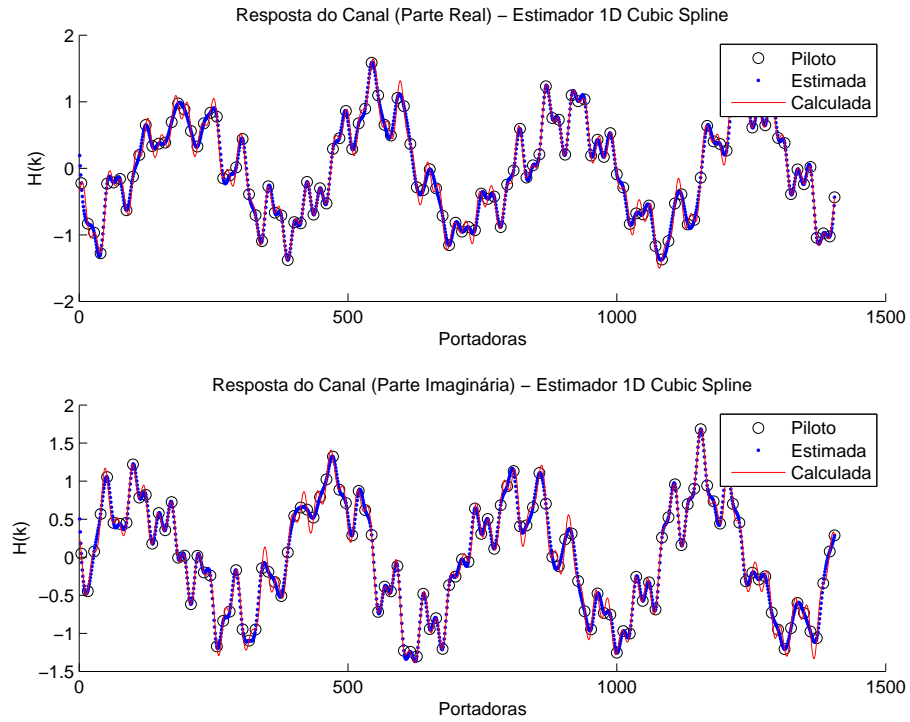


Figura 5.5: Resultado do estimador de canal 1D *Cubic-Spline* para 1 símbolo OFDM em modo 1 (Canal Brasil B).

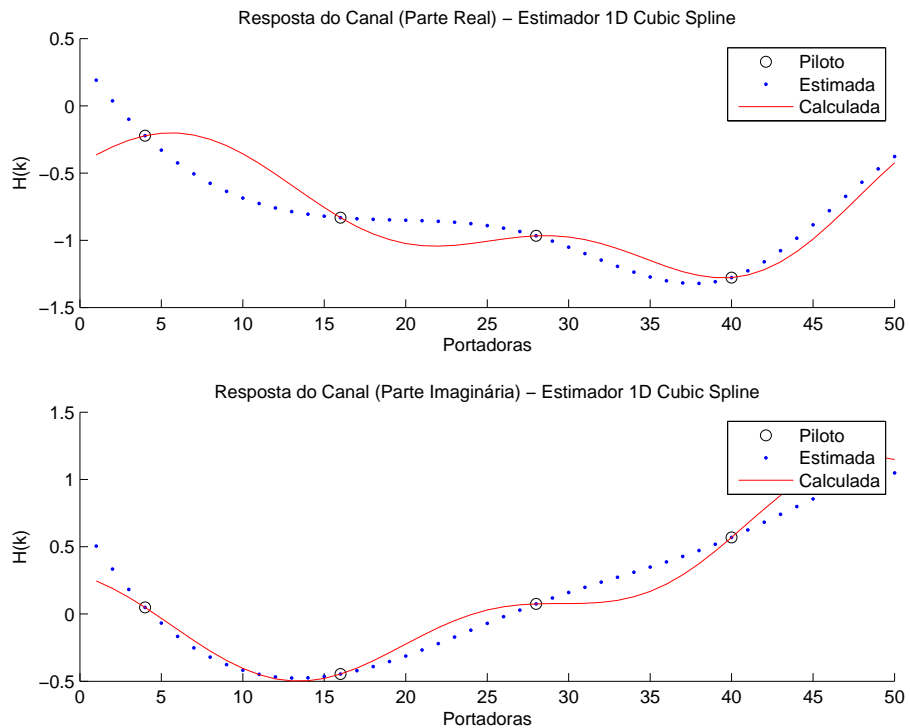


Figura 5.6: Resultado do estimador de canal 1D *Cubic-Spline* para as 50 primeiras sub-portadoras de 1 símbolo OFDM em modo 1 (Canal Brasil B).

a interpolação seja precisa. Por menor que seja o erro de estimação, os símbolos QAM ainda irão se encontrar deslocados de sua posição ideal, como ilustra a Figura 5.7.

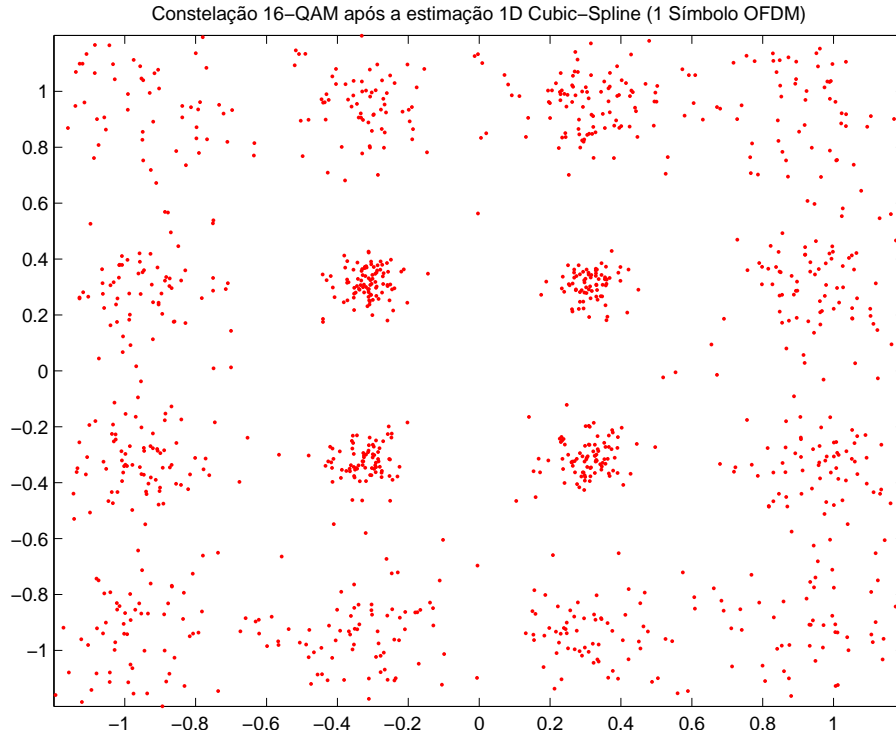


Figura 5.7: Constelação 16-QAM das subportadoras de 1 símbolo OFDM com estimação 1D *Cubic-Spline* corrompida por um canal de desvanecimento Rayleigh (Brasil B).

A mesma análise pode ser feita para o método de estimação em duas dimensões. As Figuras 5.8 e 5.9 mostram os resultados da estimação de canal para 1 símbolo OFDM inteiro e para as 50 primeiras subportadoras do mesmo símbolo respectivamente.

Analisando a Figura 5.9, visualiza-se que os erros de interpolação foram significativamente reduzidos pela estimação de canal 2D. Essa melhoria é atribuída à redução de espaçamento entre as portadoras de referência. A constelação corrigida com o método de estimação 2D é mostrada na Figura 5.10.

Pode-se notar que a estimação 2D Cubic Spline corrigiu todos os danos causados pelo canal ao símbolo OFDM analisado. Porém, na prática isso não ocorre. Devido os danos do canal estarem acompanhados de ruído, o desempenho do estimador cai consideravelmente devido ao ruído das portadoras de referência. Esse erro pode ser visto analisando a Figura 5.11, que trás o resultado da estimação 2D (50 primeiras portadoras) para o mesmo cenário que os resultados anteriores, porém com uma relação sinal-ruído de 12 dB.

Essa queda de desempenho do estimador provoca, em canais AWGN, uma queda de

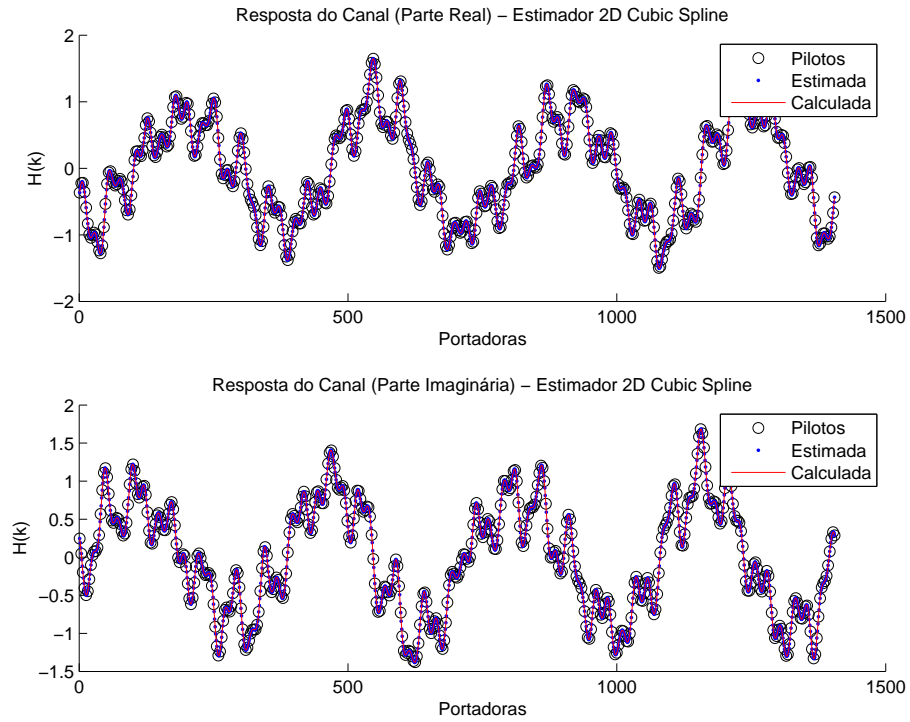


Figura 5.8: Resultado do estimador de canal 2D *Cubic-Spline* para 1 símbolo OFDM em modo 1 (Canal Brasil B).

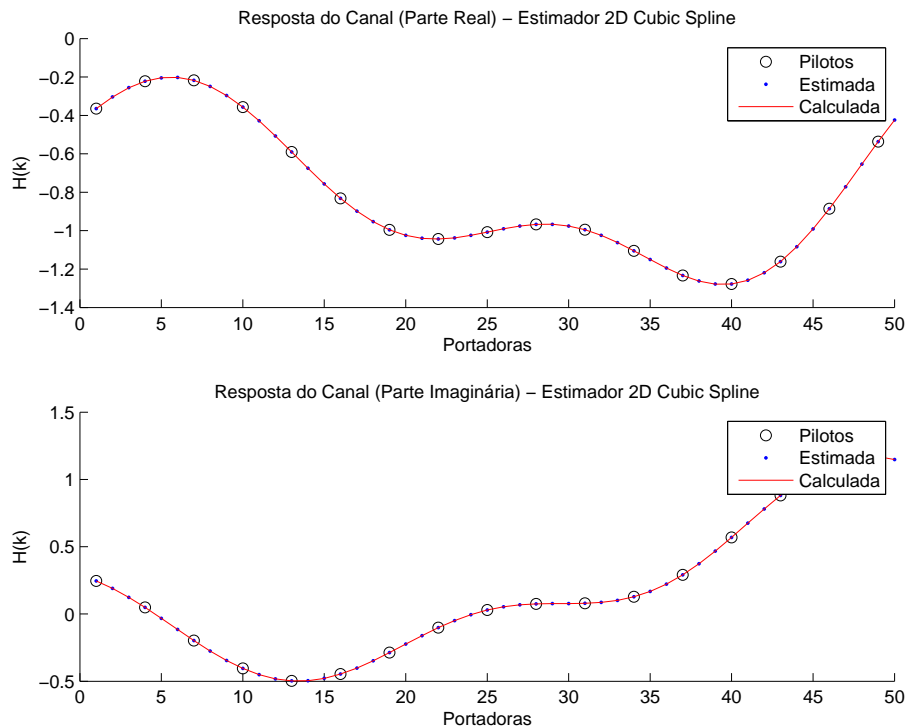


Figura 5.9: Resultado do estimador de canal 2D *Cubic-Spline* para as 50 primeiras sub-portadoras de 1 símbolo OFDM em modo 1 (Canal Brasil B).

desempenho do sistema de aproximadamente 2 dB. Essa queda de desempenho pode ser vista analisando a Figura 5.12.

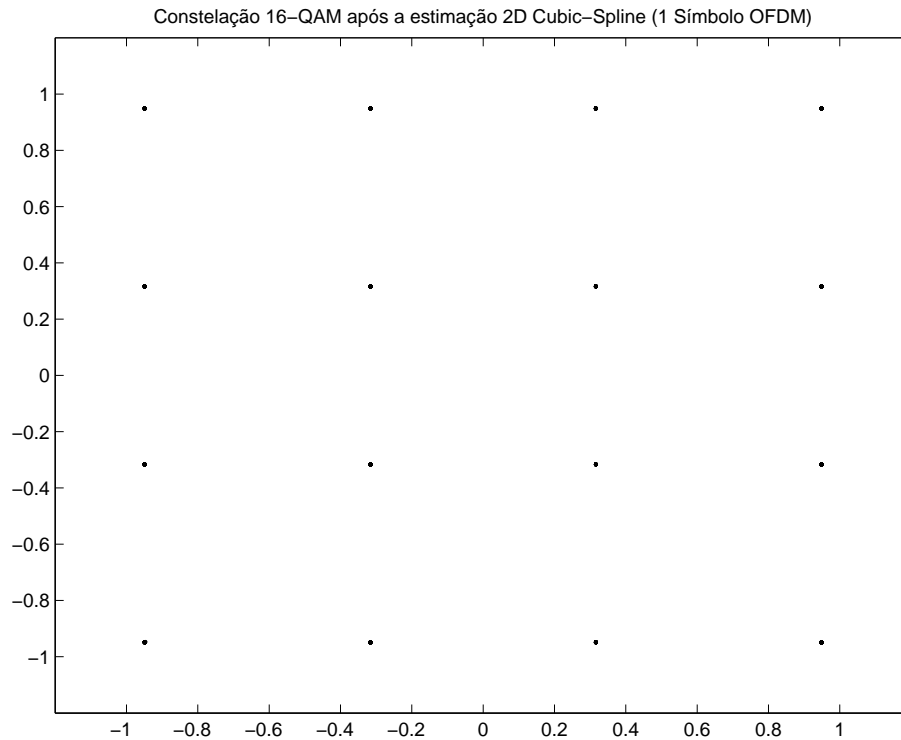


Figura 5.10: Constelação 16-QAM das subportadoras de 1 símbolo OFDM com estimação 2D *Cubic-Spline* corrompida por um canal de desvanecimento Rayleigh (Brasil B).

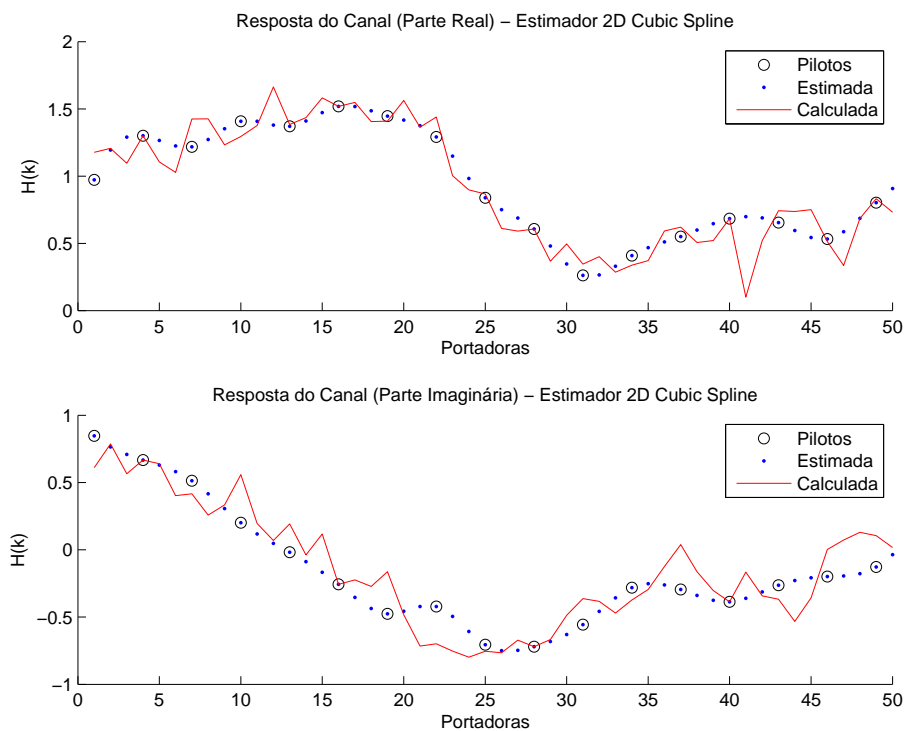


Figura 5.11: Constelação 16-QAM das subportadoras de 1 símbolo OFDM com estimação 2D *Cubic-Spline* corrompida por um canal de desvanecimento Rayleigh (Brasil B) + 12dB de ruído AWGN.

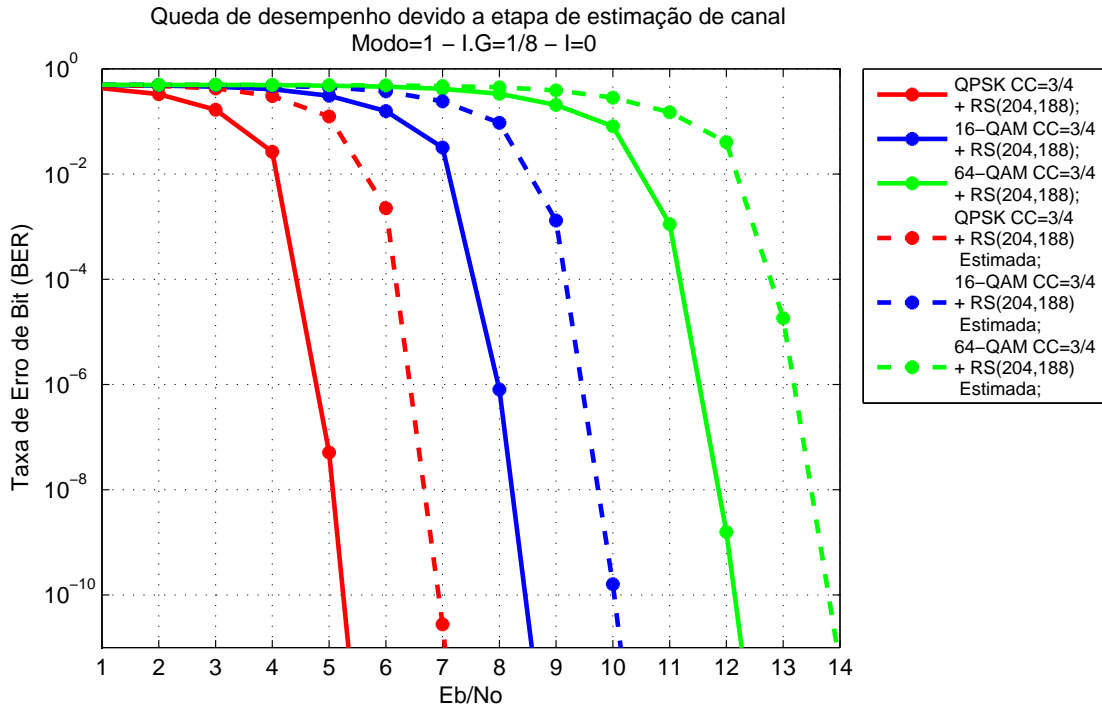


Figura 5.12: Queda de desempenho nas constelações decorrentes de erros de estimação em canais com desvanecimento + ruído AWGN.

A análise feita nesta seção pode ser estendida para outras configurações de símbolos OFDM, bem como outros modelos de canais.

## 5.4 Curvas de desempenho

Atualmente no Brasil, a maior parte das transmissoras estão optando por manterem parâmetros fixos no transmissor. A configuração de segmentos utilizada é a configuração 2, em que o segmento central é utilizado para recepção parcial e os demais para transmissão em alta definição. Os parâmetros utilizados são:

- Modulação: 64-QAM;
- Modo de operação: 3;
- Taxa do código convolucional: 3/4;
- Intervalo de guarda: 1/8 ou 1/16;



- Profundidade de entrelaçamento temporal (I): 0;

O entrelaçamento temporal apenas é utilizado em regiões que apresentem forte influência de ruído impulsivo. Evita-se utilizar tal entrelaçador devido ao *delay* causado na recepção do sinal, que é fortemente percebido quando o tele-espectador realiza uma troca de canal. Pode-se dizer que em uma avaliação subjetiva do impacto do *delay* na troca de canal, o resultado será pior do que a presença de alguns artefatos na imagem decorrentes de ruído impulsivo no sinal recebido.

Essa configuração é mantida fixa, mesmo se o vídeo a ser transmitido não for um vídeo de alta definição. Neste caso, é feita uma conversão de SDTV para HDTV, resultando em um “falso” HD.

Para os parâmetros acima listados, podem-se obter as seguintes taxas de transmissão [4]:

- HDTV (Modulação 64-QAM; CC=3/4; IG=1/8): 18,25 Mb/s;
- HDTV (Modulação 64-QAM; CC=3/4; IG=1/16): 19,33 Mb/s;

Alterando as modulações, pode-se obter as taxas de transmissão para os demais tipos de vídeos:

- EDTV (Modulação 16-QAM; CC=3/4; IG=1/8): 12,17 Mb/s;
- EDTV (Modulação 16-QAM; CC=3/4; IG=1/16): 12,89 Mb/s;
- SDTV (Modulação QPSK; CC=3/4; IG=1/8): 6,08 Mb/s;
- SDTV (Modulação QPSK; CC=3/4; IG=1/16): 6,44 Mb/s;

As curvas de desempenho (BER x  $E_b/N_0$ ), geradas pelo simulador, têm papel fundamental para atingir o objetivo proposto por este trabalho. Através destas curvas pode-se observar os impactos das variações de parâmetros do sistema de maneira clara e objetiva, verificando o ganho na robustez da transmissão.

Analisando a Figura 5.13, é possível visualizar o desempenho de uma transmissão em HDTV, com os parâmetros utilizados atualmente pelas transmissoras.

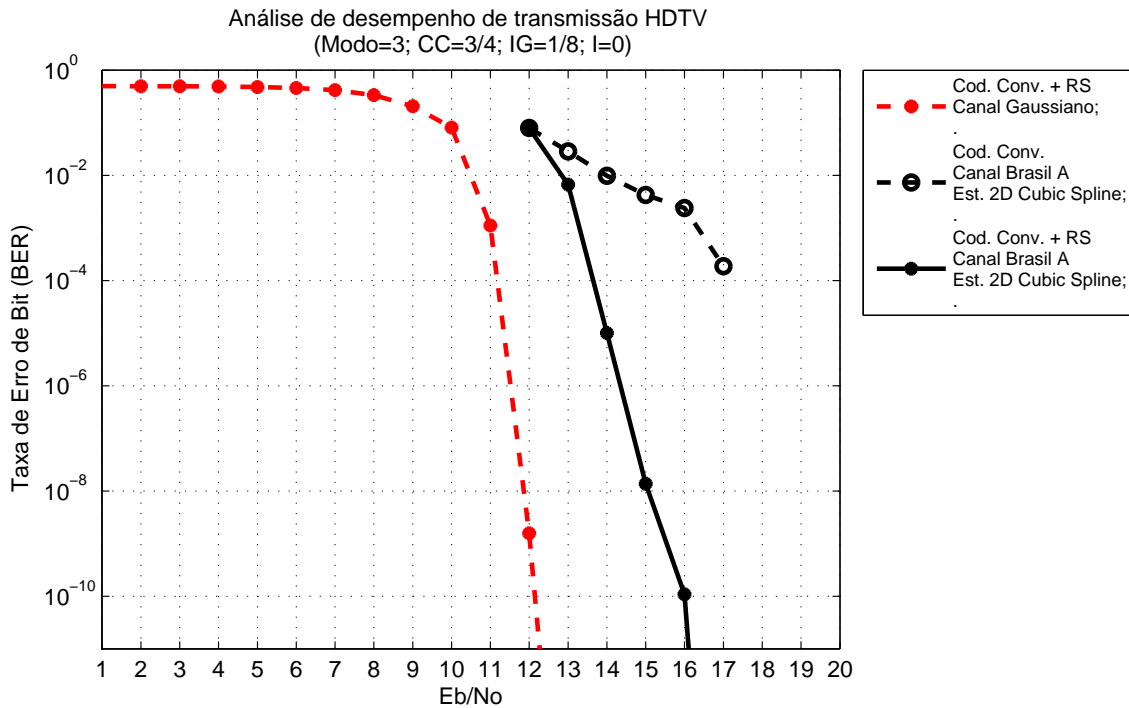


Figura 5.13: Exemplo de desempenho de uma transmissão em HDTV.

Os resultados de uma transmissão em EDTV é dado pela Figura 5.14.

Por fim, a mesma transmissão em SDTV tem como resultado as curvas trazidas pela Figura 5.15.

Comparando o desempenho das transmissões analisando as Figuras 5.15 e 5.13, pode-se notar um ganho significativo de robustez na transmissão. Na prática esse ganho pode ser aplicado no caso de parametrizar o transmissor de forma correta quando o vídeo transmitido não estiver em alta definição. Assim, o processo de “conversão” de SDTV para HDTV seria poupado e o sinal ficaria mais imune às interferências do canal.

Em [28], na seção *Analysis of covering area for DTV standards*, são apresentadas as áreas de cobertura que podem ser obtidas pelo transmissor ISDB-T para uma transmissão quase livre de erros, como mostra a Tabela 5.1.

Extrapolando os resultados trazidos por [28], pode-se obter um ganho em área de cobertura, se o transmissor for parametrizado levando em consideração o tipo de mídia

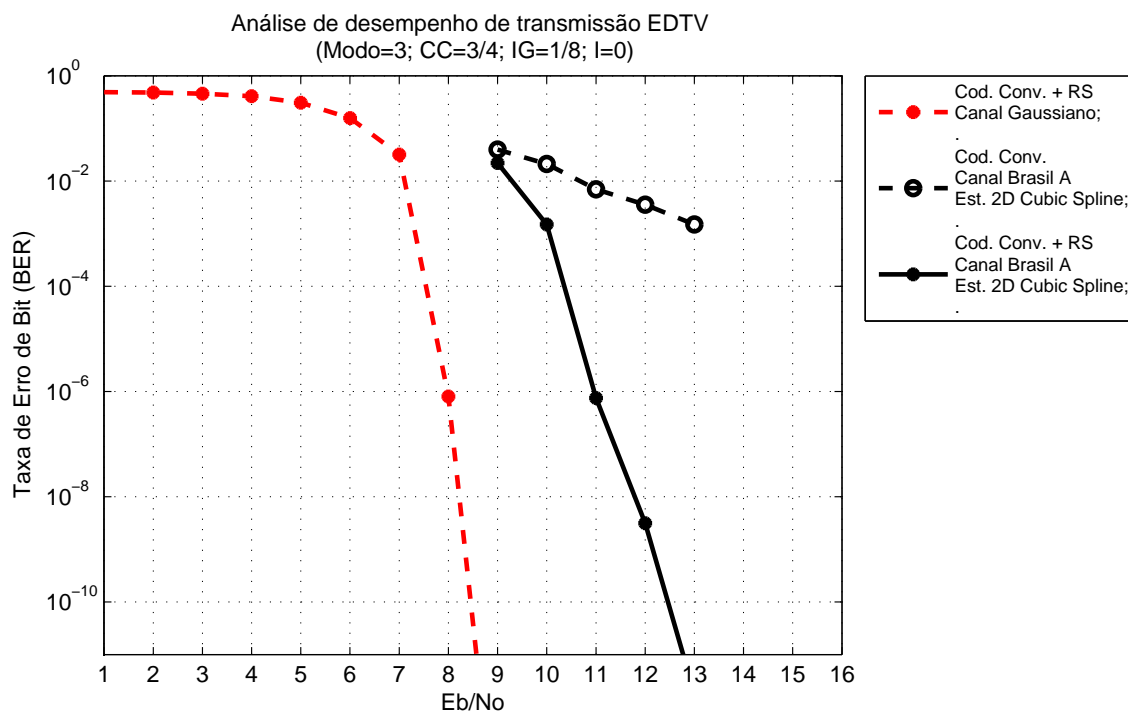


Figura 5.14: Exemplo de desempenho de uma transmissão em EDTV.

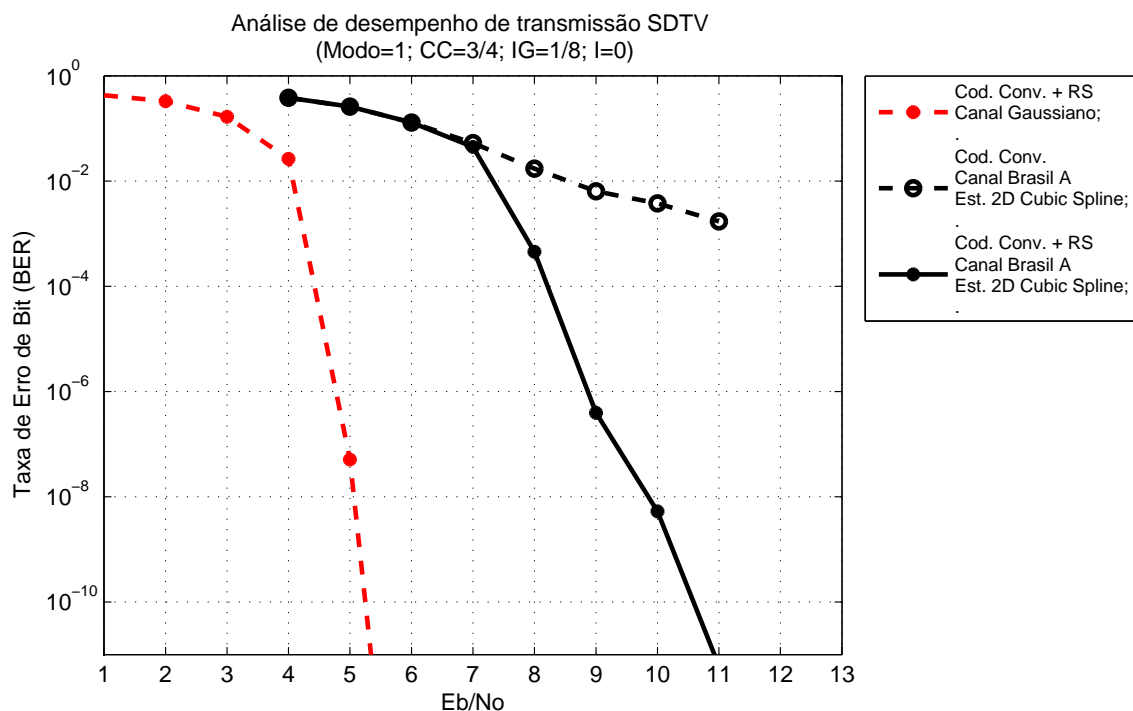


Figura 5.15: Exemplo de desempenho de uma transmissão em SDTV.

que se deseja transmitir. Com o transmissor operando com parâmetros para realizar uma transmissão em alta definição, o simulador encontrou uma SNR de aproximadamente 22 dB para uma recepção quase livre de erros. O que resulta em uma área de cobertura

Tabela 5.1: Áreas de cobertura para o sistema ISDB-T.

	VHF	UHF
HDTV (SNR de 20,1 dB)	56 Km	28 Km
EDTV (SNR de 14,6 dB)	67 Km	35 Km
SDTV (SNR de 6,6 dB)	84 Km	46 Km

um pouco menor que 56 Km para VHF e 28 Km para UHF. Caso a média que se deseja transmitir seja do tipo SDTV (SNR de aproximadamente 13 dB obtida pelo simulador), e se o transmissor for parametrizado para esse tipo de transmissão, pode-se obter um ganho em área de cobertura maior que 11 Km para VHF e maior que 7 Km para UHF.

Através de simulações, outras comparações podem ser feitas, como por exemplo, alterações no intervalo de guarda. Desta forma, uma previsão do desempenho da transmissão pode ser feita ao se consultar a base de dados.

Para apresentar os resultados o simulador gera duas figuras distintas. Uma para camadas individuais e outra com os resultados das transmissões de todas as camadas. A primeira figura contém: a curva teórica de desempenho da modulação sem codificação de canal, a curva simulada para a mensagem codificada com o código convolucional e a curva de desempenho do código convolucional concatenado com o código RS(204,188). As curvas de desempenho em canais AWGN vistos nas Figuras 5.15, 5.14 e 5.13 foram incluídos apenas neste trabalho, não sendo traçadas naturalmente pelo simulador.

Na segunda figura gerada pelo simulador, são apresentados somente os resultados da curva de desempenho do código convolucional concatenado com o código RS(204,188), porém para as três camadas (quando utilizadas).

## 5.5 Resultados complementares

Com objetivo didático, são gerados algumas figuras que visam facilitar o entendimento de algumas partes do sistema ISDB-T. São elas: figura dos TSPs remultiplexados, figura da distribuição de segmentos e representação gráfica da inserção do intervalo de guarda.

Também são gerados resultados que buscam mostrar os impactos sofridos pela transmissão de forma amigável: vídeos das constelações e imagens transmitidas (TIFF ou PNG).

### 5.5.1 Figura da distribuição de segmentos e intervalo de guarda

A Figura 5.16 ilustra qual é a distribuição dos segmentos utilizadas na simulação, nesse caso é a configuração 2. Outra informação trazida por esta figura é a taxa do intervalo de guarda adicionado à frente do símbolo OFDM.

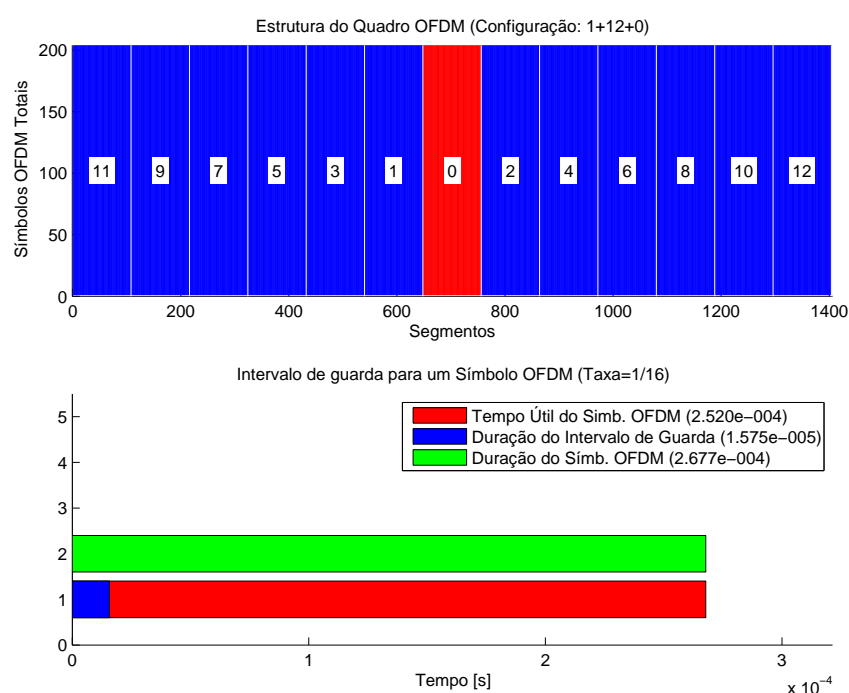


Figura 5.16: Ilustração da distribuição dos segmentos OFDM nas camadas hierárquicas para a configuração 2 e inserção do intervalo de guarda.

### 5.5.2 Figura dos TSPs remultiplexados

Quando é feita a transmissão de imagens pelo simulador, o bloco de remultiplexação do simulador é utilizado. A Figura 5.17 ilustra a distribuição dos TSPs em suas camadas, tornando possível visualizar a distribuição dos pacotes ao longo do TS remultiplexado. Note que a camada de maior capacidade possui mais TSPs que as camadas com menos segmentos. Outra informação que pode-se tirar desta figura é a resolução das imagens remultiplexadas.

No exemplo da Figura 5.17, foi utilizada a configuração 3 (transmissão nas três camadas hierárquicas). A camada C possui uma imagem TIFF com 512x512 pixels, contudo, as imagens das camadas de menor capacidade, tiveram reduzidas suas quantidades de pixels, visando utilizar uma mesma quantidade de quadros para transmitir as imagens de todas as camadas.

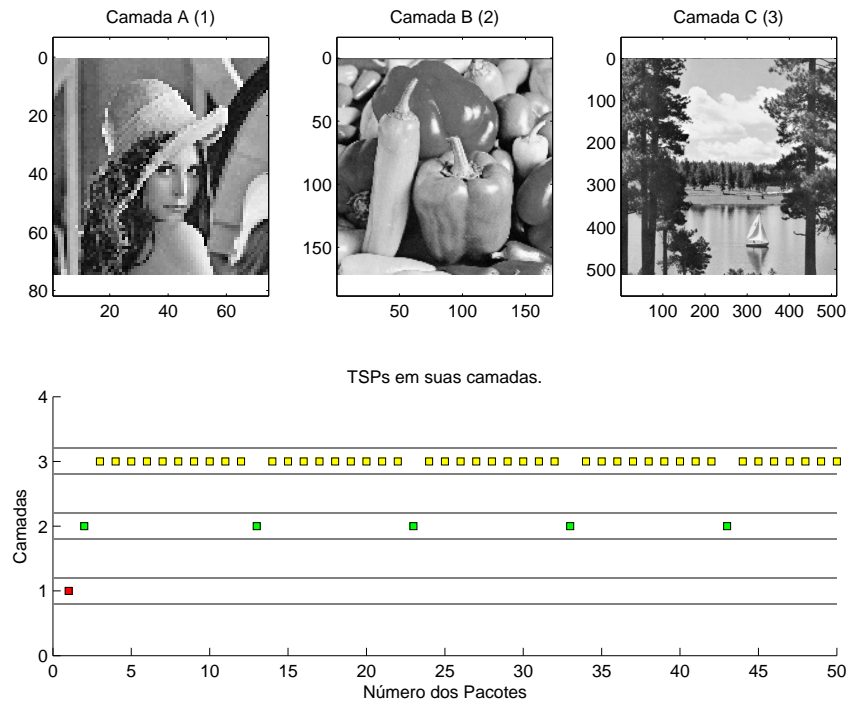


Figura 5.17: TS remultiplexado com os dados das imagens TIFF para a configuração 3.

### 5.5.3 Imagens transmitidas

A Figura 5.18 traz um exemplo de imagem PNG recebida utilizando o modelo de canal Brasil A ( $E_b/N_0 = 5\text{dB}$ ). Foram necessários 3 quadros OFDM, com arredondamento no último, para concluir a transmissão (6% de uma imagem de resolução 1920x1080x3). Os parâmetros utilizados para realizar a transmissão desta imagem são: Modulação QPSK, intervalo de guarda 1/8, taxa de codificação igual a 3/4,  $I = 0$  em modo 3.

Note que o sistema ISDB-T<sub>b</sub> utiliza codificação de vídeo H.264/AVC, desta forma, a imagem PNG não irá apresentar efetivamente os artefatos que o vídeo transmitido teria na prática. Para ter uma visão mais realista, seria necessário transmitir um vídeo codificado.

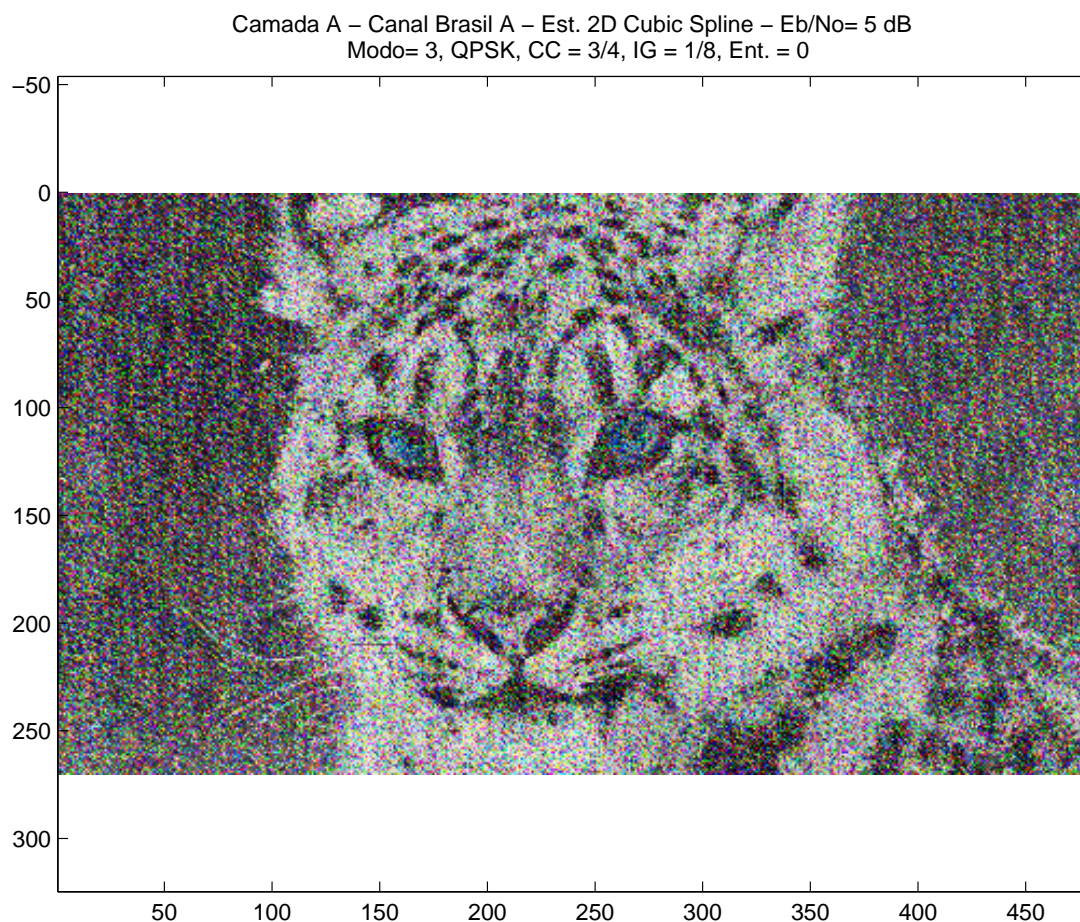


Figura 5.18: Exemplo de imagem recebida em 3 quadros OFDM, configuração 1, modo 3, modulação QPSK, intervalo de guarda de 1/8 e código convolucional puncionado para 3/4.

Porém, a imagem transmitida pode ser utilizada, pelo usuário do software de visualização de resultados, com o objetivo de facilitar a interpretação das curvas de desempenho apresentadas. Tais imagens dão uma idéia visual do impacto do ruído ao sinal transmitido.

#### 5.5.4 Vídeos das constelações

Os dados para geração de vídeos são salvos em arquivos de texto para serem utilizados pelo software de visualização de resultados. Uma idéia de como as constelações são mostradas pelo software de visualização de resultados será dada no Capítulo 6.

## 5.6 Considerações finais

Neste capítulo foram apresentados os resultados obtidos através do simulador desenvolvido neste trabalho. Vale lembrar que além dos resultados mostrados neste capítulo, o simuladora gera um arquivo de *log* da simulação, que pode ser utilizado para facilitar ainda mais o entendimento dos blocos do sistema.

Os resultados aqui mostrados podem ser estendidos para a maioria das combinações de parâmetros do sistema. No capítulo seguinte será visto como o software de visualização de resultados irá mostrar os resultados gerados pelo simulador, bem como sua estrutura de funcionamento.



## CAPÍTULO 6

---

### O software criado

---

#### 6.1 Introdução

O software de visualização de resultados, desenvolvido em Delphi, recebe o nome de “ISDB-T Analyser”. Compilado para plataforma Windows, o software expõe os resultados das simulações da seguinte forma:

- Curvas de desempenho da simulação (BER x  $E_b/N_0$ );
- Imagens TIFF/PNG recebidas para cada nível de ruído;
- Vídeos das constelações para cada nível de ruído;
- Estrutura do TS remultiplexado;
- Configuração do quadro OFDM;
- Representação gráfica da inserção do intervalo de guarda;

- Relatório contendo os passos realizados pelo simulador.

Tal software está preparado para receber a estrutura de arquivos criada pelo simulador. Essa estrutura é criada de maneira automatizada, garantindo que o usuário não precise estruturar cada simulação manualmente.

O simulador tem capacidade de gerar vídeos das constelações no formato AVI (*Audio Video Interleave*), porém, isso se mostrou muito ineficiente do ponto de vista de espaço em disco. Desta forma, o software gera os vídeos das constelações através de um vetor de dados salvo dentro das pastas de simulações, isso para cada nível de ruído.

O ISDB-T Analyser possui um sistema de importação de simulações, onde outras simulações poderão ser feitas posteriormente, até mesmo por outros trabalhos, e terem seus resultados incluídos no software.

## 6.2 Estrutura geral

De maneira geral, a estrutura do ISDB-T Analyser é bem simples, e pode ser analisada com o auxílio do fluxograma da Figura 6.1. Esse fluxograma é simplificado, onde ficaram de fora blocos de verificação de existência de arquivos, blocos de avisos, blocos de *skins*, importação de simulações e blocos de opções extras.

Ao iniciar o programa, o software realiza uma busca por simulações e lista as encontradas. Caso não encontre nenhuma, avisa ao usuário da necessidade de importar as simulações. Caso encontre arquivos de simulações já importados, os parâmetros destas simulações serão incluídos nas listas de parâmetros disponíveis para consulta.

Em seguida, o programa aguarda a seleção dos parâmetros desejados, e quando requisitado, o programa busca por simulações que contenham tais parâmetros. Caso a busca retorne vários resultados, uma lista de simulações é criada para que o usuário selecione a desejada. Após selecionada, os dados da simulação são carregados e os botões de visualização de resultados são habilitados. Quando pressionados, os botões mostram as janelas

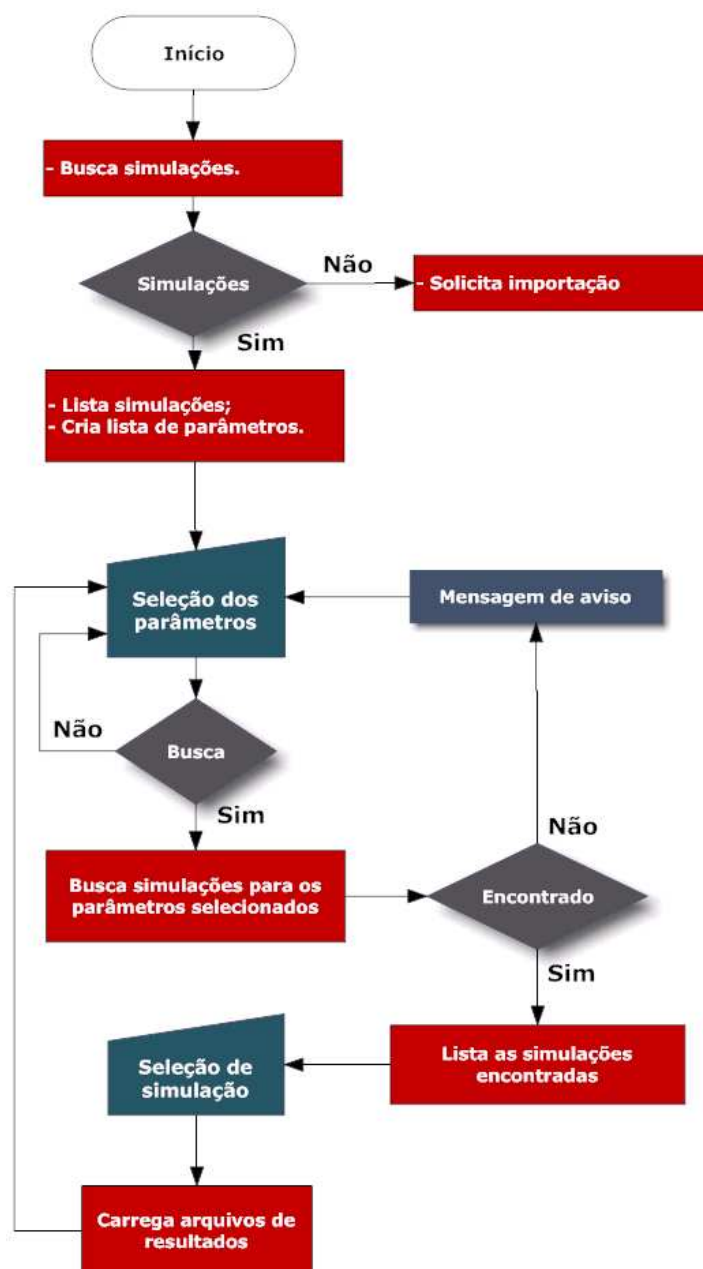


Figura 6.1: Fluxograma simplificado do software de visualização de resultados.

responsáveis por exibir o resultado desejado.

Como o objetivo do software é auxiliar os profissionais que trabalham com os transmissores, eventualmente acadêmicos ou pessoas que não dominam a teoria do sistema por completo, deseja-se que este tenha arquivos de ajuda com comentários, dicas e explicações para auxiliar no entendimento dos resultados consultados. Com esse fim, uma base de dados de informações foi criada de modo que possam ser inseridos comentários e sugestões dos próprios usuários do software.

Essa base de dados de informações possui uma interface independente do software de visualização de resultados e utiliza Access como base de dados DB (*Data Base*). A escolha deste tipo de base de dados é justificada pelo fato de não ser necessário a instalação de nenhum tipo de software extra para acessar o conteúdo do DB, sendo que o Delphi possui *drivers* destinados para leitura e escrita deste tipo de base de dados.

Por fim, deve-se considerar que o software necessita possuir um instalador para facilitar a utilização e distribuição do mesmo. Após realizar a instalação, o usuário pode importar arquivos de simulações e visualizar os resultados de maneira simplificada e com boa interface gráfica.

### 6.3 O ISDB-T Analyser

Ao iniciar o programa a primeira imagem que o usuário irá visualizar é trazida pela Figura 6.2. A primeira barra de status indica o andamento da varredura das pastas de simulações importadas anteriormente. Após a varredura ser concluída, o programa irá processar os dados de configuração encontrados na busca anterior. Esse processamento de dados consiste em selecionar os parâmetros de cada simulação encontrada, eliminar os parâmetros repetidos e adiciona-los às caixas de seleção. Desta forma, apenas os parâmetros que se encontram efetivamente simulados e importados irão aparecer como opção dentro do programa.

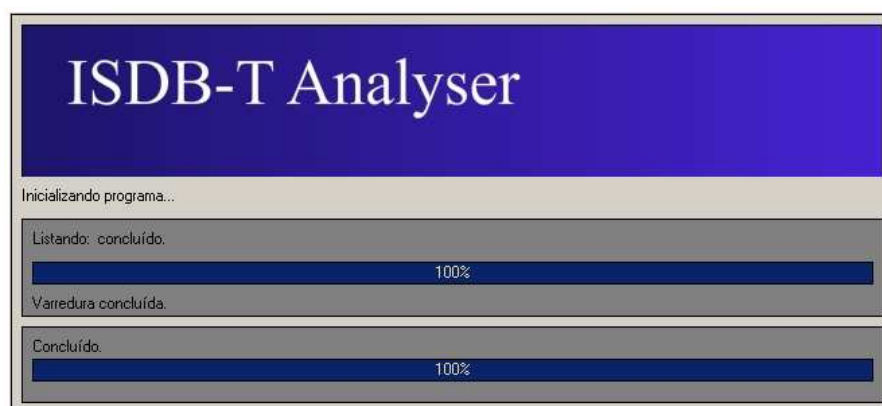


Figura 6.2: Janela de *splash* do software de visualização de resultados.

Caso o programa não encontre nenhuma simulação, ele sugere a importação das mesmas. Se o usuário optar por importá-las, ele será conduzido a uma janela igual à ilustrada pela Figura 6.3. Após a seleção do caminho onde se encontra a pasta contendo as simulações, deve-se clicar em buscar simulações. Desta forma, o programa irá localizar as pastas que contenham simulações válidas, ou seja, cuja estrutura de arquivos contidos nestas pastas estejam de acordo com a estrutura requerida pelo programa.

Se não for a primeira vez que o usuário estiver importando simulações, após localizá-las, o programa irá verificar pastas de simulações duplicadas. Caso já existam simulações com os mesmos parâmetros que as simulações encontradas, o software pergunta ao usuário se o mesmo deseja remover as simulações anteriormente importadas. Essa etapa do processo é mostrada na Figura 6.3.

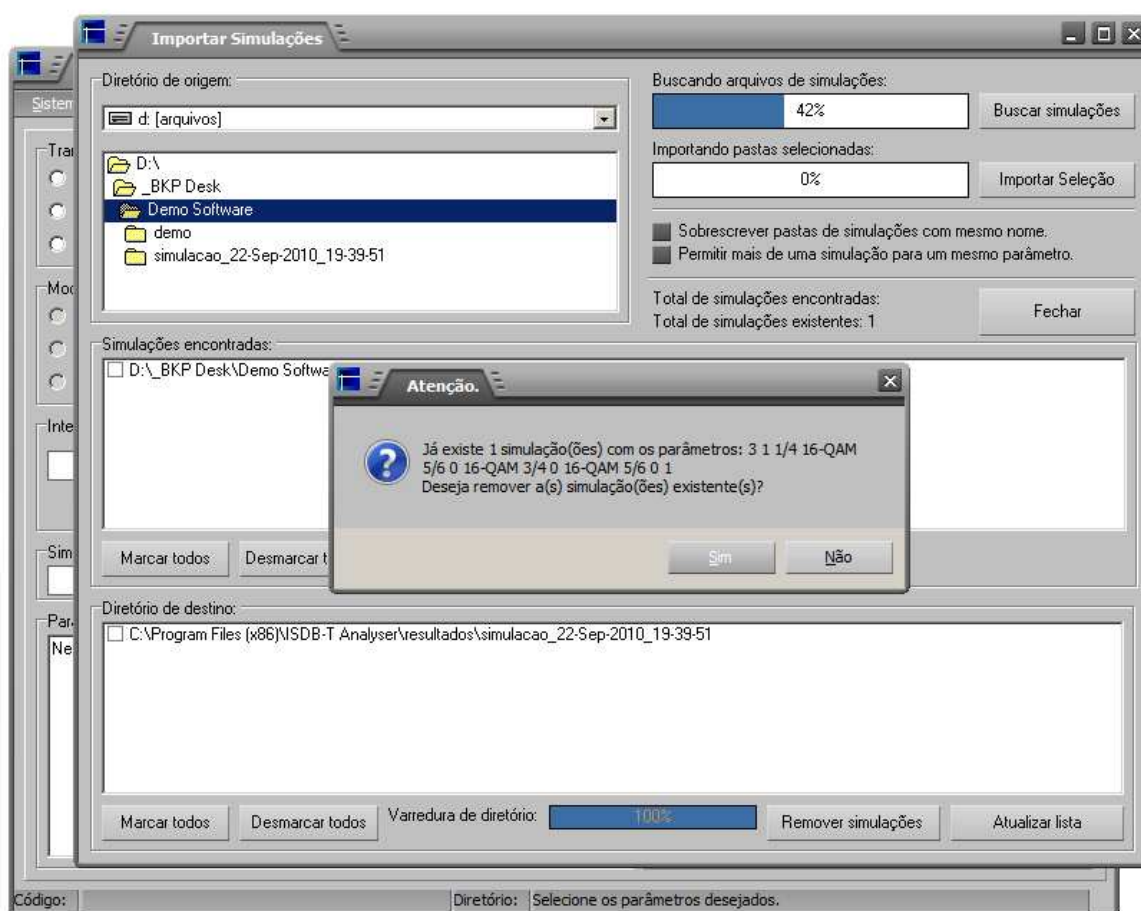


Figura 6.3: Janela de importação de simulações do software de visualização de resultados.

Considerando que o processo de importação de simulações já tenha sido concluído, após a janela da Figura 6.2, o usuário irá visualizar a janela ilustrada pela Figura 6.4. Essa é a janela principal, onde o usuário pode selecionar os parâmetros das simulações que deseja localizar. Se o usuário preferir, ele pode selecionar a simulação na barra de ferramentas, selecionando a opção “lista de simulações”. Essa última opção facilita o uso do programa no caso do sistema possuir poucas simulações.

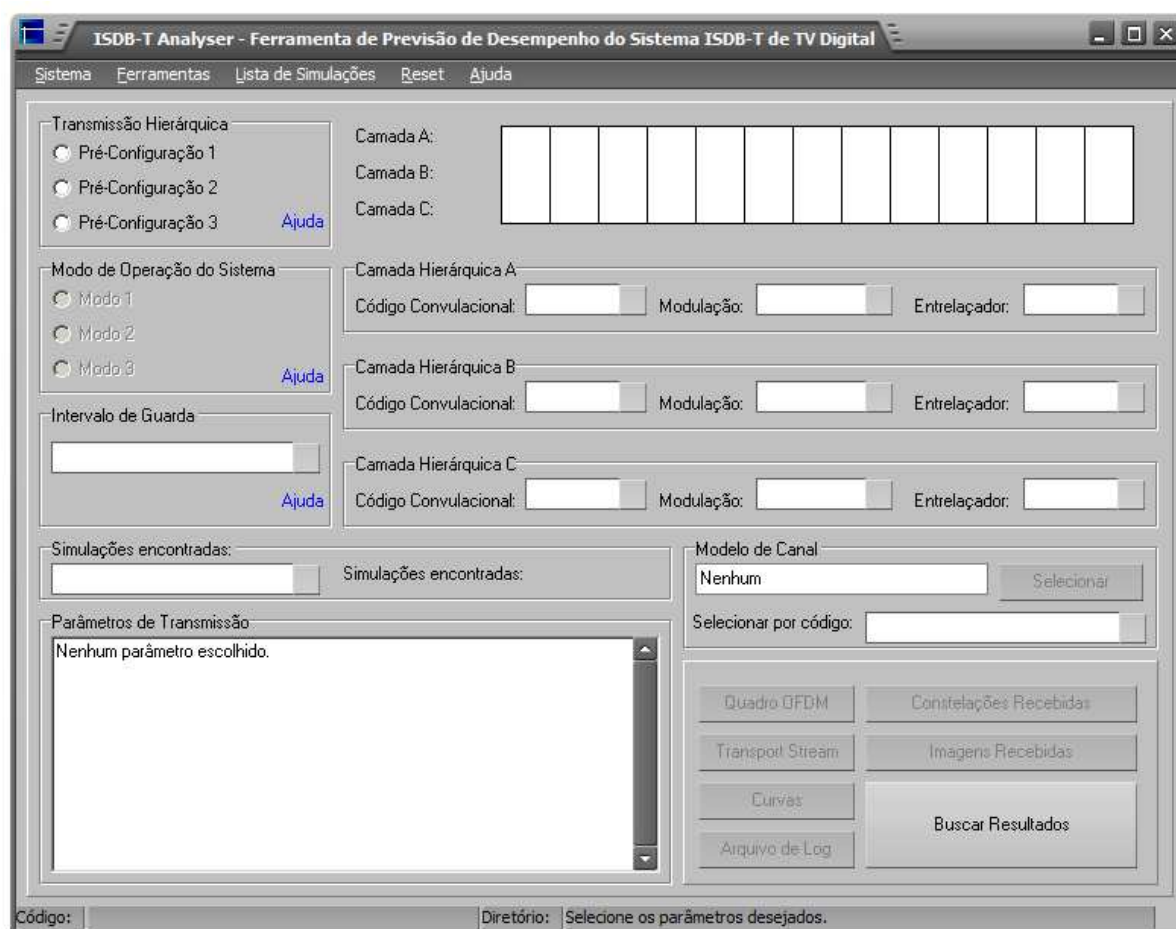


Figura 6.4: Janela principal do software de visualização de resultados.

Após feita a seleção dos parâmetros, basta clicar no botão “Buscar Resultados” e o programa irá listar as simulações encontradas para os parâmetros entrados, como mostra a Figura 6.5. Esse parâmetros selecionados na ilustração da Figura 6.5 são fictícios e não representam uma configuração comumente usada.

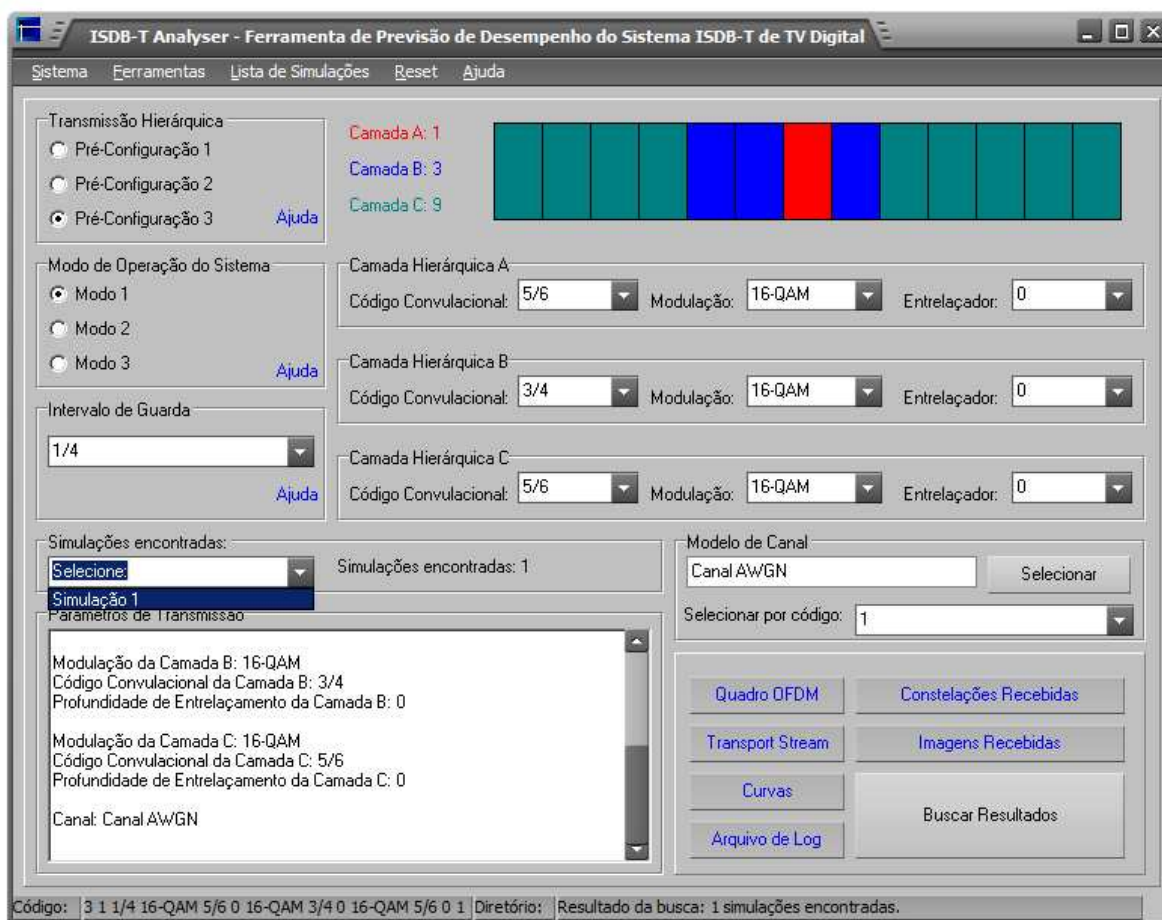


Figura 6.5: Janela de principal após seleção dos parâmetros e conclusão da busca.

Após selecionar uma simulação dentre as encontradas, os botões de visualização de resultados são liberados. Desta forma, o usuário estará livre para ver os resultados da simulação selecionada.

Clicando em “Imagens Recebidas”, o usuário poderá visualizar o resultado da transmissão de imagens pelo sistema ISDB-T com os parâmetros selecionados, como mostra a Figura 6.6.



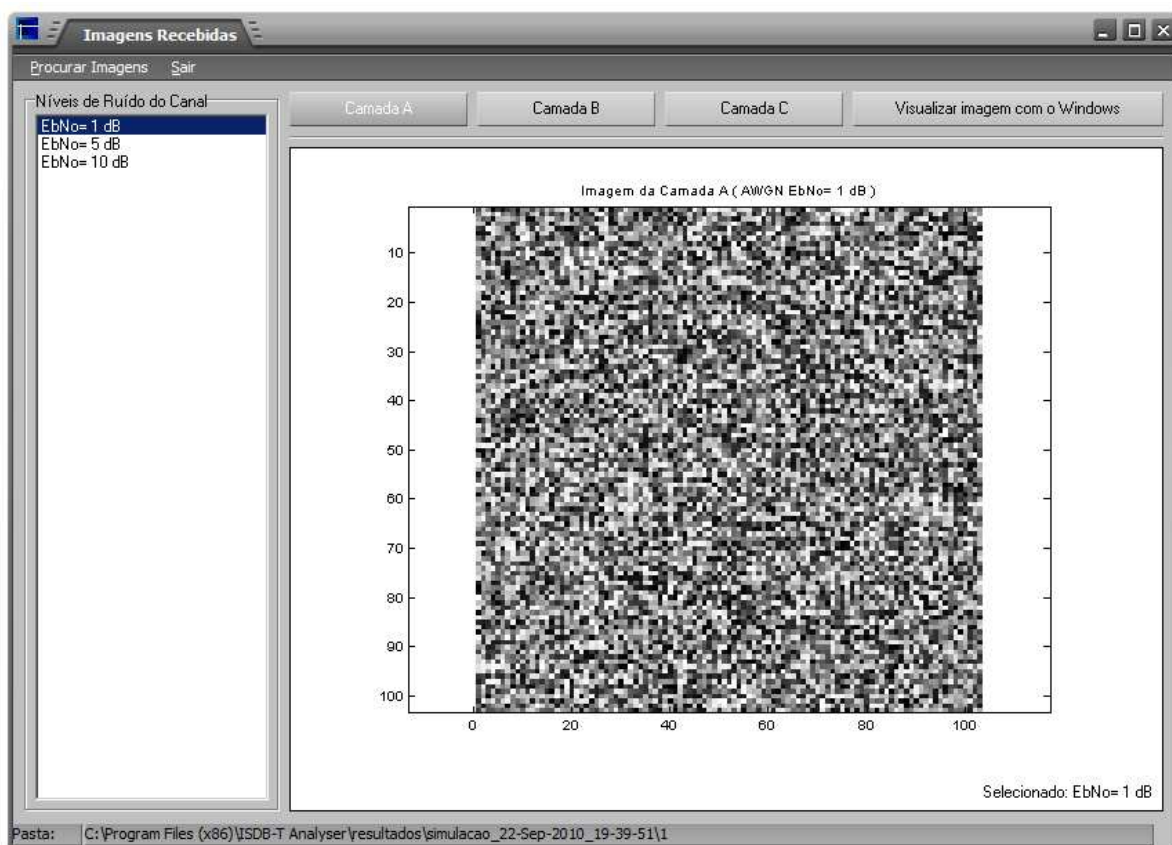


Figura 6.6: Janela de visualização de imagens recebidas do software de visualização de resultados.

Voltando à janela principal, ao clicar em “constelações recebidas”, uma janela igual a da Figura 6.7 irá aparecer. Essa janela é responsável por gerar os vídeos das constelações de cada uma das camadas, sendo essas estimadas ou não. Para gerar os vídeos, são necessários converter os dados salvos nos arquivos de textos em coordenadas numéricas do quadro de reprodução. Essa conversão pode ser feita em tempo real ou com um esquema de pré-processamento dos dados. Por padrão, os dados são tratados em tempo real, porém, caso o computador no qual o software se encontra instalado não tenha recursos para realizar o processamento dos dados em tempo real, pode-se selecionar a opção “pré-processamento” para preparar os dados antes da reprodução do vídeo. Outro recurso de reprodução do vídeo é o “zoom”, que pode ser ajustado pelo usuário da maneira que ele preferir.

Cada ponto do vetor de símbolos, gerado pelo simulador, é um ponto na constelação.



Desta forma, o usuário pode selecionar a quantidade de pontos por quadro que serão plotados e a quantidade de quadros por segundo, alterando a velocidade de reprodução do vídeo. O tamanho do ponto na imagem também pode ser alterado.

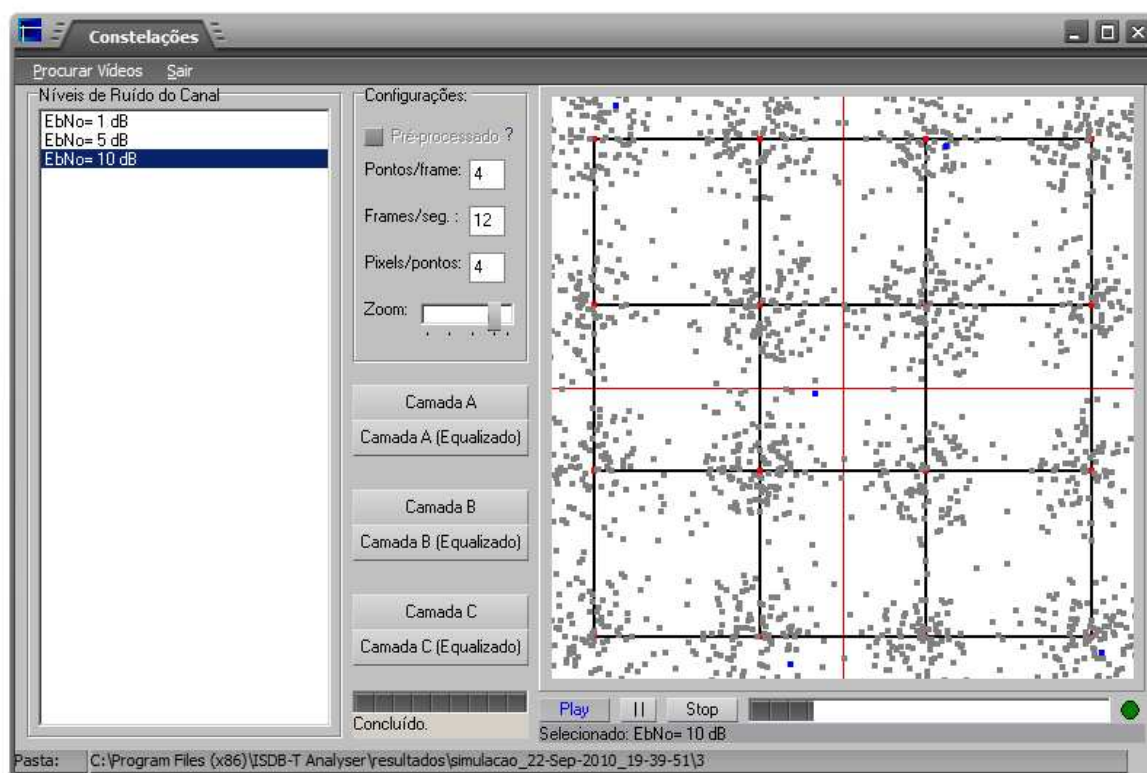


Figura 6.7: Janela de geração do vídeo das constelações recebidas do software de visualização de resultados.

Retornando à janela principal, caso o usuário clique na opção “Quadro OFDM”, uma janela igual a da Figura 6.8 irá aparecer. Essa janela irá mostrar a imagem da distribuição dos segmentos no quadro OFDM e a adição do intervalo de guarda ao símbolo. No canto direito da janela, o usuário pode consultar as descrições e comentários deixados pelos usuários a respeito destes parâmetros. Essas informações se encontram no banco de dados do programa, e podem ser alteradas através do editor de informações a qualquer momento.

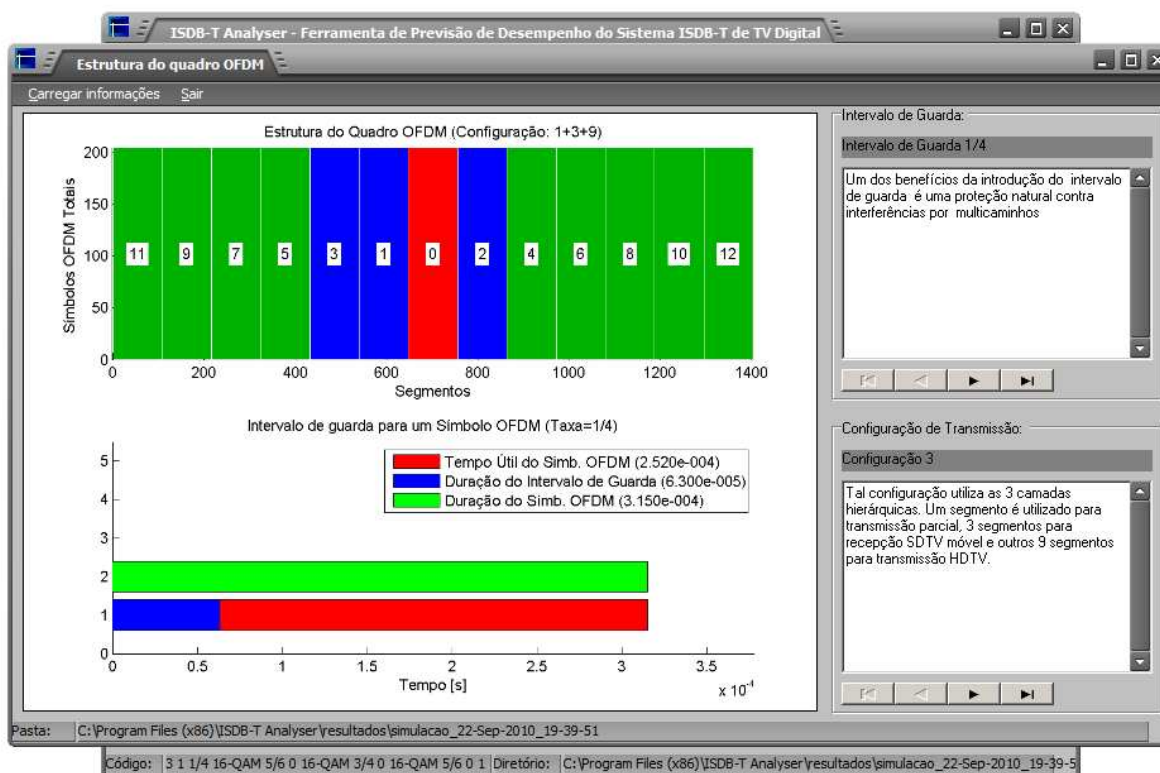


Figura 6.8: Janela de visualização da distribuição de segmentos do quadro OFDM e adição do intervalo de guarda do software visualização de resultados.

A mesma estrutura de janela é usada para mostrar as informações do TS remultiplexado gerado pelo simulador através das imagens TIFF. Basta clicar em “Transport Stream”, na janela principal do programa, e uma estrutura igual a da Figura 6.9 irá aparecer.



Figura 6.9: Janela de visualização da estrutura do TS remultiplexado do software de visualização de resultados.

O Software conta com algumas opções extras, como por exemplo, tabela de portadoras de controle, *skins* e botões de “ajuda rápida”. Indo em “ferramentas”, no menu superior, e clicando em “Tipos de portadoras” o usuário poderá ver a quantidade de portadoras totais necessárias para compor um quadro OFDM com os parâmetros selecionados por ele. Essa opção, juntamente com a opção de “ajuda rápida”, podem ser vistas analisando a Figura 6.10. Nesta figura, o usuário selecionou o botão de ajuda rápida da tabela de portadoras.

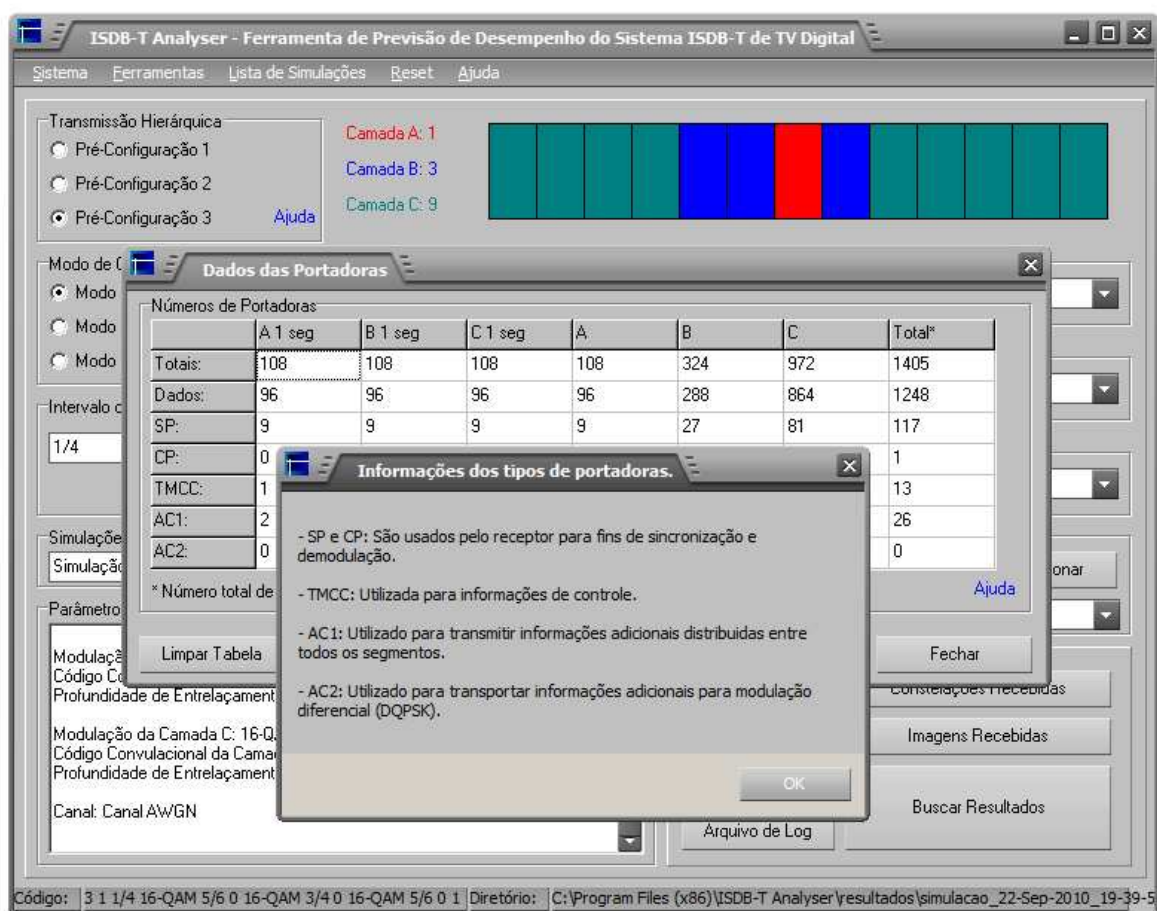


Figura 6.10: Consulta ao botão de “ajuda rápida” da janela de descrição de portadoras.

Uma visão geral do software de visualização de resultados foi dada, onde alguns recursos como: arquivo de ajuda, menu de *skins*, botão “sobre o software” entre outros, foram poupadas visando tornar o capítulo mais sucinto.

Por fim, a Figura 6.11 mostra o editor de informação do banco de dados do programa. Esse editor é totalmente independente do software principal, possui suas funções exclusivamente para gerenciamento do DB. Selecionando o tópico desejado, o usuário pode acrescentar, excluir ou editar informações que serão consultadas através do software principal. Esse recurso visa auxiliar ambientes onde existem várias pessoas com acesso ao software, permitindo uma troca de informações e experiências entre elas.

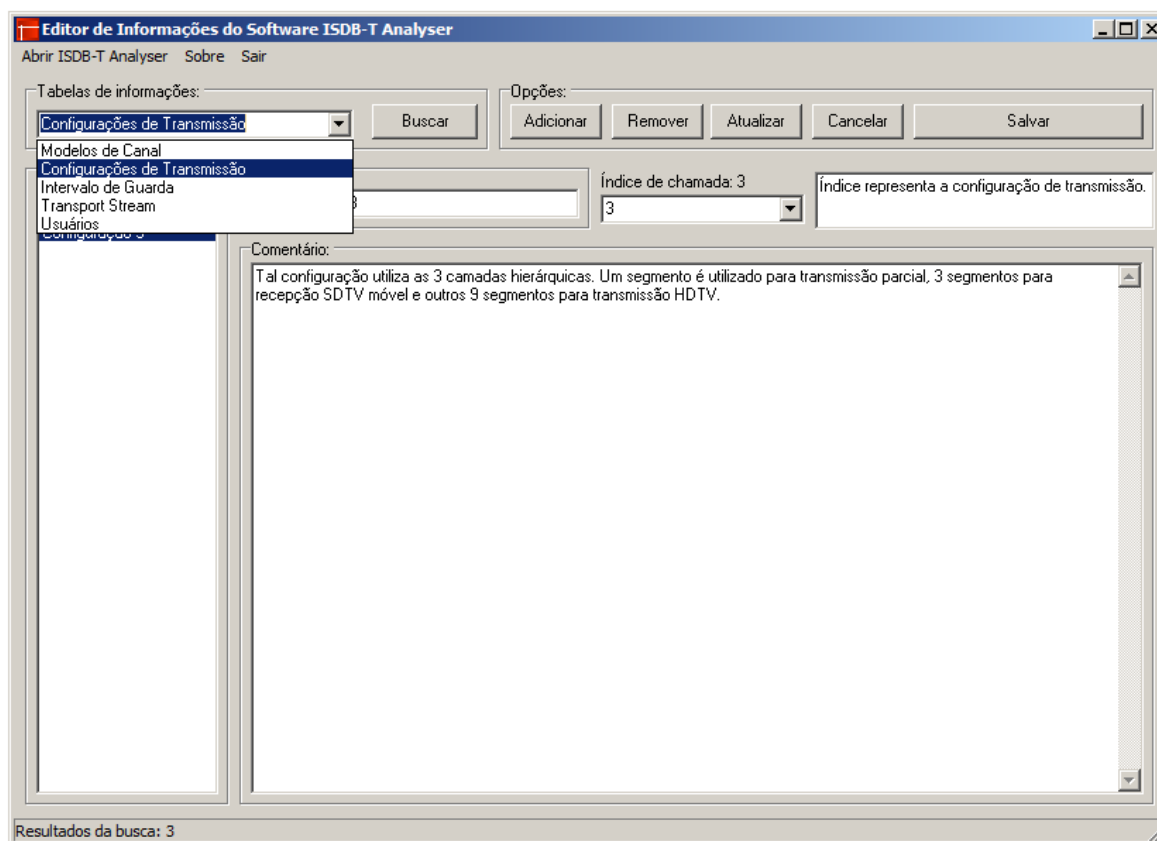


Figura 6.11: Janela principal do editor de informações de banco de dados do software de visualização de resultados.

## 6.4 Vantagens e Desvantagens

Uma dúvida que pode ser levantada é: Qual a vantagem de ter um sistema de análise de resultado baseado em uma base de dados? Ou ainda: Por que não gerar as simulações em tempo real? O principal problema de gerar os resultados em tempo real é a demora das simulações, quais podem levar dias para serem concluídas. Isso irá depender da quantidade de informação que se deseja transmitir pelo simulador.

Utilizando um computador com processador Intel Core i3 (M330) com dois núcleos de processamento de 2.13GHz cada e 4GB de memória RAM (*Random Access Memory*), para os parâmetros:

- Uma camada hierárquica;

- Dados gerados aleatórios (9 milhões de bits);
- Sem entrelaçador temporal ( $I=0$ );
- Código convolucional de taxa  $3/4$ ;
- 16 níveis diferentes de ruído;
- Canal AWGN;

o tempo de simulação varia de 16 minutos (modulação 64-QAM) à 1 hora (Modulação QPSK), pois em modulações com menor quantidade de bits por símbolo, são gerados mais quadros OFDM, o que acresce o tempo de simulação.

Em simulações com leitura de imagens, para as mesmas condições listadas acima, o tempo de simulação para modulação QPSK leva em torno de 2 horas e meia para ser concluída. Quando o simulador utiliza a leitura de imagens, são acrescentadas outras rotinas no simulador, como já discutido no Capítulo 4, resultando em um aumento no tempo de simulação.

O tempo de simulação pode se estender ainda mais. Se for utilizado um modelo de canal do tipo Brasil, onde a estrutura de repetição “loop média” (vista no Capítulo 4) será maior que um, a simulação pode levar mais do que um dia para ser concluída.

Esse tempo pode ser reduzido se as simulações forem rodadas em um *cluster* de processamento. Contudo, mesmo reduzindo significativamente o tempo de processamento, a geração de resultados em tempo real ainda será um problema, pois irá ser necessário fazer uso de computadores com grande capacidade de processamento.

Uma desvantagem do sistema proposto é o espaço em disco necessário para armazenar os resultados das simulações. No caso de simulações em tempo real, esse espaço pode ser considerado nulo.

Em uma simulação com:

- Uma camada hierárquica;

- Dados transmitidos: Aproximadamente 5,5 milhões de bits (Imagem PNG 640x360x3);
- 20 níveis diferentes de ruído;
- Canal AWGN (“loop média”=1).

serão necessários aproximadamente 20 MB de espaço em disco. Caso essa simulação seja feita com um dos modelos de canais Brasil, com “loop média” igual a 10 repetições, o espaço em disco irá subir para aproximadamente 210 MB.

O ISDB-T Analyser tem a opção de mostrar os resultados das simulações sem estimador de canal. Para isso, o simulador deve estar habilitado para salvar os resultados sem estimação. Caso essa opção esteja habilitada, serão necessários aproximadamente 416 MB de espaço em disco.

Desta forma, ao gerar as simulações, deve-se levar em consideração a necessidade de visualizar os resultados sem estimação (vídeos das constelações e imagens recebidas). Em um ambiente mais acadêmico, esta opção pode ser interessante.

Vale apenas citar que quanto maior for a imagem transmitida, maior será o espaço em disco necessário para armazenar os resultados.

Caso seja desejada uma maior economia de espaço em disco, pode-se utilizar a transmissão de dados aleatórios, onde não serão salvos nenhuma imagem. Assim, o espaço em disco irá cair para aproximadamente 90 MB mesmo com o “loop média” igual a 10 repetições e dados das constelações não estimadas salvos. Em um caso de grande economia de espaço, pode-se salvar apenas as curvas de desempenho, onde o espaço em disco ficará em torno de 275KB para qualquer tipo de simulação.

## 6.5 Considerações finais

Um ponto importante a ser considerado é a total independência entre o simulador e o software de visualização de resultados. Isso torna possível a geração de resultados por

parte de vários grupos de estudos, os quais podem compartilhar os resultados obtidos e inserir comentários à base de dados através do editor, garantindo a troca de informação e o aperfeiçoamento dos resultados em questão.



## CAPÍTULO 7

---

### Conclusões

---

Este trabalho apresentou uma solução relativamente simples de previsão de desempenho do sistema ISDB-T para diferentes configurações do transmissor. No Capítulo 2 foram revisados dois conceitos importantes: sistema OFDM e canais de radiodifusão.

No Capítulo 3 uma abordagem sucinta das várias etapas de tratamento da informação que o sistema ISDB-T possui foi feita. Nesta abordagem pode-se verificar que esse sistema tem capacidade de operar com várias configurações de transmissão. Para entender como os resultados das simulações são gerados, a metodologia do simulador criado em Matlab foi descrita no Capítulo 4. Foi possível verificar as etapas de tratamento da informação implementadas em Matlab, bem como a forma que o sistema utiliza para integrar os resultados gerados com o software de visualização de resultados.

A forma de visualização de resultados, juntamente com alguns exemplos destes, foram vistos no Capítulo 5. A estrutura de resultados gerados pelo simulador foi apresentada juntamente com alguns exemplos de curvas. Um exemplo bem prático de como o software proposto pode ser útil para as transmissoras também foi comentado.

Por fim, no Capítulo 6 o princípio de funcionamento do software de visualização de resultados foi descrito. Foram vistas também as janelas apresentadas aos usuários e algumas ferramentas de auxílio trazidas pelo software (editor de banco de dados, tópicos de ajuda, entre outros).

Com relação ao desempenho do sistema, percebe-se que o esquema de concatenação dos códigos corretores de erros dão ao sistema grande capacidade de correção de erros, que combinado com os entrelaçadores, proporcionam ao ISDB-T um bom desempenho em condições diversas de operação.

Os demapeadores com método de decisão suave aumentam consideravelmente a capacidade de correção de erros do algoritmo de Viterbi (decodificação interna), tornando a implementação deste método de decisão necessária para a obtenção das taxas de erro de bit mostradas neste trabalho.

É possível concluir que um bom sistema de estimação de canal é fundamental para garantir a equalização do sinal recebido. Sem um sistema de estimação de canal adequado, a transmissão se torna inviável. Apesar de não se ter conhecimento de quais métodos de estimação são utilizados na prática, decorrente dos projetos de receptores serem de responsabilidade das empresas, o sistema de estimação de canal em duas dimensões com método de interpolação Cubic Spline apresentou bom desempenho frente ao espalhamento temporal. Contudo, sabe-se que em simulações com efeito Doppler (recepção móvel), a estimação em uma dimensão apresenta melhores resultados.

Foi possível observar que a operação do sistema ISDB-T (transmissor) com parâmetros fixos (configuração para vídeos em HD) perde bastante em robustez se não for levado em consideração o tipo de sinal que se deseja transmitir (SDTV, EDTV ou HDTV). Essa questão se torna bastante relevante quando se sabe que no Brasil nem todos os programas são gravados em HD.

Os parâmetros que representam os valores de atenuações e atrasos dos modelos de canais Brasil, estabelecidos pela ITU-R (*International Telecommunication Union - Radiocommunication Sector*), criam várias condições de recepção que efetivamente são en-

frentadas pelos receptores. Isso dá aos resultados gerados pelo simulador criado, mais proximidade com a realidade.

Um diferencial do simulador criado está na inclusão de todos os blocos parametrizáveis do sistema ISDB-T, salvo as considerações feitas na Seção 4.2, juntamente com os modelos de canais Brasil com um esquema de estimação de canal eficiente. Os resultados obtidos por esse trabalho mostram que é possível ter uma boa estimativa de desempenho do sistema ISDB-T<sub>b</sub> através de simulações. Por isso, um grande diferencial perante os trabalhos relacionados na Seção 1.2, está justamente na geração de resultados do sistema que se encontra em operação atualmente, utilizando modelos de canais criados para representar as condições de transmissão do território brasileiro.

Em suma, o ISDB-T Analyser cumpriu com o proposto por este trabalho. Os resultados das simulações são consultados facilmente sem qualquer dificuldade de operação. Além das curvas de desempenho, a visualização das constelações e imagens recebidas proporcionam uma boa sensação visual dos impactos causados ao sinal decorrente do canal de transmissão.

## 7.1 Sugestões para trabalhos futuros

Um caminho que pode ser tomado a partir deste trabalho é a transcrição do código fonte do ISDB-T Analyser para uma linguagem de ambiente de programação livre. Assim, poderão ser criadas versões do software para plataforma Linux, por exemplo. Da mesma forma, o simulador pode ser implementado em outra linguagem de programação visando um ganho nos tempos de simulação, como por exemplo uma implementação em C.

Outro item para implementação futura é a adição, junto ao software de visualização de resultados, de um mecanismo que possa passar para o usuário o efeito do *delay* causado pelo processo de entrelaçamento temporal. Tornando a ferramenta mais interessante para operadores do transmissor. Ainda com relação ao entrelaçamento temporal, pode-se implementar no simulador um bloco de adição de ruído impulsivo, como em [26].

Isso resultará em uma melhoria na análise dos ganhos de robustez do sinal causado pela profundidade de entrelaçamento.

Outras implementações como: etapas de sincronismo do quadro OFDM, leitura das TMCCs e um sistema de decisão suave para a modulação DQPSK, podem ser perseguidas. Assim, resultados para transmissão móvel podem ser gerados e acrescentados ao ISDB-T Analyser.

Um outro tipo de estrutura que o sistema proposto pode adotar é a disponibilidade de uma versão *online* do ISDB-T Analyser. Assim, a base de dados com resultados estaria hospedada em um servidor, que pode receber resultados de simulações gerados por outros pesquisadores da área.

---

## Bibliografia

---

- [1] G1.com. Brasil tem mais tv do que geladeira, diz estudo. <http://g1.globo.com>. Abril de 2007.
- [2] dvb.org. <http://www.dvb.org>. Setembro de 2010.
- [3] Estevan Marcelo Lopes. Técnicas de alocação adaptativa do comprimento dos filtros digitais de equalizadores para sistemas de hdtv no padrão atsc. Dissertação de Mestrado, Instituto Nacional de Telecomunicações, 2002.
- [4] Abnt nbr 15601:2007.
- [5] Estadão. Tv digital da china vai movimentar 62,5 bilhões de dólares em 2015. <http://www.estadao.com.br>. Setembro de 2006.
- [6] Dayan Adionel Guimarães e Geraldo Gil Ramundo Gomes. Caracterização do canal para o mi-sbtvd. páginas 14, 2006.
- [7] Instituto de Estudos e Projetos em Comunicação e Cultura. <http://www.indecs.org.br>. Junho de 2006.

- [8] Cristiano Akamine. Análise de desempenho do sistema de tv digital dvb-t com entrelaçador temporal em canais com ruído impulsivo. Dissertação de Mestrado, Universidade Estadual de Campinas - Pós-Graduação em Engenharia Elétrica - Departamento de Telecomunicações, Campinas - SP, 2004.
- [9] Diogo Gará Caetano. Análise de desempenho de sistemas de tv digital móveis dvb-h e isdb-t utilizando concatenação de códigos bch e turbo codificadores. Dissertação de Mestrado, Universidade Estadual de Campinas - Pós-Graduação em Engenharia Elétrica - Departamento de Telecomunicações, 2007.
- [10] Ricardo Gonçalves Trentin. Técnicas de processamento mimo-ofdm aplicadas à radiodifusão de televisão digital terrestre. Dissertação de Mestrado, Universidade Federal de Santa Catarina, 2006.
- [11] Rangel Arthur. *Novas Propostas para Otimização de Receptores de TV Digital Baseados em OFDM em Ambientes de Redes de Frequência Única Regionais*. Tese de Doutorado, Universidade Estadual de Campinas - Faculdade de Engenharia Elétrica e de Computação, 2007.
- [12] Marcelo Carneiro de Paiva. Uma implementação em software do subsistema de transmissão do padrão isdb-tb. Dissertação de Mestrado, Instituto Nacional de Telecomunicações, Março de 2010.
- [13] Xinrong Wang. Performance evaluation of the isdb-t standard for multimedia services. Dissertação de Mestrado, The University of British Columbia - Department of Electrical and Computer Engineering, 2002.
- [14] Luís Eduardo Antunes de Resende. Desenvolvimento de uma ferramenta de análise de desempenho para o padrão de tv digital isdb-t. Dissertação de Mestrado, Pontifícia Universidade Católica do Rio de Janeiro, 2004.
- [15] K. Fazel e S. Kaiser. *Multi-Carrier and Spread Spectrum Systems From OFDM and MC-CDMA to LTE and WiMAX*. John Wiley and Sons, Ltd, Publication, segunda edição, 2008.

- [16] Ernesto Leite Pinto e Claudio Penado de Albuquerque. A técnica de transmissão ofdm. *Revista Científica Periódica - Telecomunicações*, 5(1):10, Junho de 2002.
- [17] Bernard Sklar. *Digital Communications: Fundamentals and Applications*. Second Edition, second edition edition, 2001.
- [18] Simon Haykin. *Communication Systems*. John Wiley & Sons, Inc., 4th edition edition, 2001.
- [19] Borko Furht e Syed Ahson. *Handbook of Mobile Broadcasting - DVB-H, DMB, ISDB-T, AND MEDIAFLO*. Internet and Communications. Auerbach Publication, 2008.
- [20] Shu Lin e Daniel J Costello. *Error Control Coding: Fundamentals And Applications*. Franklin F. Kuo, 1983.
- [21] Andrea Goldsmith. *Wireless Communication*. Cambridge University Press., 2005.
- [22] Arib std-b31 version 1.6.
- [23] Makoto Sasaki. Technologies and services of digital broadcasting - terrestrial digital television broadcasting. *CORONA publishing co, Ltd, and NHK*, 20:4, 2004.
- [24] João Terêncio Dias. Desempenho de algoritmos para estimação de parâmetros de sincronização em sistemas ofdm. Dissertação de Mestrado, Instituto Militar de Engenharia, Praça General Tibúrcio, 80 - Praia Vermelha, Rio de Janeiro - RJ, 2006.
- [25] Iuri Rodrigues Baran. Explorando a coerência temporal e a comunicação oportunística em sistemas de comunicação móvel multiusuário. Dissertação de Mestrado, Universidade Federal de Santa Catarina, 2006.
- [26] Alexandre Kioshi Ishini e Cristiano Akamine. Técnicas de estimação de canal para o sistema isdb-tb. *Revista de Radiodifusão*, 3:6.
- [27] Luciano L. Mendes, José Marcos C. Brito, Fabbryccio A. Cardoso, Dayan A. Guimarães, Gustavo C. Lima, Geraldo G. R. Gomes, Dalton S. Arantes, e Richard D.

- Souza. Mi-sbtvd: A proposal for the brazilian digital television system sbtvd. *Journal of the Brazilian Computer Society*, vol.12 no.4, 2007.
- [28] Geraldo Gil Ramundo Gomes e Luciano Leonel Mendes. Covering area analysis for digital tv system. *IWT 2007 - International Workshop on Telecommunications*, 2007.